Geometry and Computation of Houndstooth (Pied-de-poule)

Loe M.G. Feijs
Department of Industrial Design
Eindhoven University of Technology
l.m.g.feijs@tue.nl

Abstract

We apply a variety of geometric and computational tools to improve our understanding of the Houndstooth (Pied de poule) pattern. Although the pattern must have been known for centuries, it was made famous mostly by Christian Dior and is still frequently used in many variations. It is a non-exhaustible source of inspiration for fashion designers.

1 Introduction

Houndstooth denotes a family of patterns. The French Pied de poule means “foot of hen”. We create computer programs generating the patterns, trying to capture the essence of Pied de poule in programs which are as compact as possible. We use these as tools to see what happens in the extreme cases of the coarsest and finest possible grids. We find for example that, in a precise sense, the English term “puppy-tooth” is more correct than the Dutch “molentjes” for the simplest possible pattern. We find, as expected, that all patterns can be described by the Heesch-Kienzle theory of tilings (as are many works by M.C. Escher). Somewhat unexpectedly, we find that the limit case has different Heesch-Kienzle type than the regular patterns.

This section introduces the origin and applications of the pattern. In Section 2 we create compact programs generating the pattern (one for each $N > 0$). In Section 2 we let N go to infinity. tessellations theory is used in Section 3. The contours of the basic figure are tackled using special turtle commands (Section 4). Section 4 gives a fresh look at the very smallest figure. Feeding the computer-generated contours into another modern tool, a laser cutter, was also a source of fresh ideas: see Section 5 for novel patterns.

Weaving is the process of creating a fabric from 2 perpendicular thread sets. One set is called the *warp* (vertical), the other the *weft* (horizontal). The weft thread crosses either over or underneath the warp threads and the precise about rule where to cross over and where to cross underneath determines the weave type of the fabric thus made. Examples are plain, twill and satin weave. In twill weaving the pattern of the warp going up and down is more involved than just one-up one-down. A frequently used twill is the 2/2 twill (two warp up and two warp down). For each subsequent weft thread, the pattern is shifted by one position. For a longer explanation see for example [1] or [2]. Twill is the preferred weave for kilts, overcoats, uniforms etc.

Let the warp threads be colored according to a simple one dimensional pattern, e.g. 4 white, 4 black, 4 white, 4 black and so on. In the same way the four white, four black repeated pattern can be used for the weft threads. In combination with the a 2/2 twill weaving, the most interesting pattern of Figure 1 appears. It is known by the French term Pied de poule, in English it is Houndstooth and in Dutch Hanevoet. Whereas printed patterns (flowers, Paisley print) are vulnerable to wear and becoming faint after washing, the woven pattern is more robust: the threads were dyed before the weaving (and printing is often one-sided). The woven pattern is sharp: the edges of the pattern coincide with transitions between threads. We conjecture this to be one reason why the Pied de poule pattern is often taken as a symbol of high quality fashion (next to the robust nature of the fabric, cf. kilts, overcoats and uniforms). Later, printed Pied de poule was produced. As shown in Figure 1 (right) it is applied in contemporary fashion (e.g. in the 2009 winter collection of the Preen label). Other labels like Moschino, DKNY, Stella McCartney, and Alexander McQueen also use it.

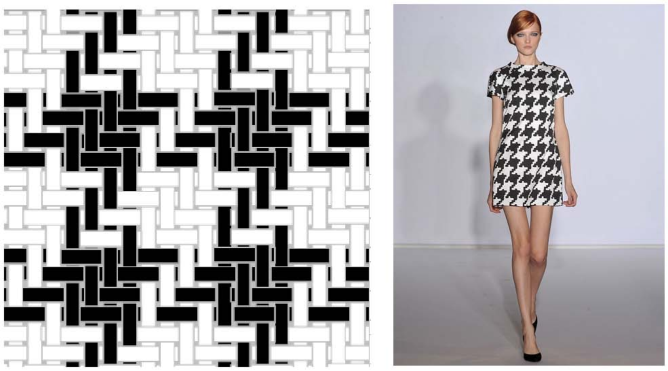


Figure 1 : *Pied de poule fabric with two up two down twill (source: wikimedia) and application (Preen autumn winter 2009) [www.elle.nl/mode/Modetrends/Wintertrend-2009-pied-de-poule].*

The warp-up warp-down effects can be simulated by a digital computer program. Our first programs mimicked the behavior of the loom, describing the construction of weft threads, going from left to right. This works (Fig. 2 left) but is impractical as the algorithm must be reworked for other patterns. Next we made table-driven programs. We provide them via the Bridges CD-rom and `www.idemployee.id.tue.nl/l.m.g.feijs/pieddepoule.html`. Two more screen shots of such program's output are in Fig. 2.

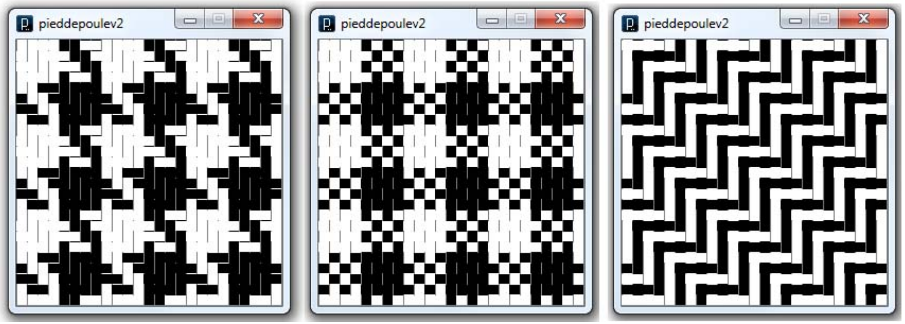


Figure 2 : *Algorithmic Pied de poule and results of data driven program*

A table-driven program uses modular arithmetic for array indices. The warp and the weft colours are stored in 1-dimensional arrays (1 is white, 0 is black). The weaving pattern is in a 4 × 4 array (or 8 × 8, the program is easily adapted). The value is 1 or 0 depending on which of warp and weft is on top. We can experiment with the patterns.

$$\text{Consider } W_1 = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{bmatrix} \text{ and let } W_2 = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

The middle pattern in Fig. 2 is from `warp` = {1,1,1,1,0,0,0,0}; `weft` = {1,1,1,1,0,0,0,0}; and `weave` = W_1. It is another checkerboard, seen in cook's clothing. The right hand side pattern in Figure 2 is from `warp` = {1,0,1,0,1,0,1,0}; `weft` = {1,0,1,0,1,0,1,0}; `weave` = W_2. It is a kind of herringbone (Dutch: visgraat).

2 Compact characterizations

Then we continued searching for even more compact characterizations. There is something very specific about the well-known Pied-de-poule, herringbone or checkerboard patterns. The warp and weft arrays are filled with simple alternations like 1 with 0 or 1,1 with 0,0 or 1,1,1,1 with 0,0,0,0. The notion of complexity as proposed by Kolmogorov is a source of inspiration. According to Kolmogorov [3], the entropy of an x is is the length of a shortest program P, itself written as as sequence of zeroes and ones, that outputs x. We should characterize our patterns by formulas that are as compact as possible.

Let $M\%m$ be the integer number x such that $0 \leq x < m$ and $x \equiv M(\bmod m)$. We define a Boolean function of coordinates, i and j, say, such that the outcome *true* means that the area designated by i and j becomes white; *false* means black. The notation ␣ ? ␣ : ␣ is if-then-else. Here is such a compact formula:

$$a_{ij} = (i-j)\%6 < 3 \ \ ? \ \ i\%12 < 6 \ : \ j\%12 < 6$$

The condition $(i-j)\%6 < 3$ determines whether the horizontal (weft) thread is chosen or the vertical thread (the warp). In the former case i determines the color, in the latter case j determines the color. The condition $(i-j)\%6 < 3$ tells that the weave pattern repeats after 6 positions. The essence is in a one-line Processing (=Java) formula, embedded in the program as the 8th line.

```
int i = 1000;
stroke(128);
boolean aij;
size(250,250);
for (int x = 0; x < 100; x++){
    int j = 0;
    for (int y = 0; y < 100; y++){
        aij = (i-j)%6<3  ?  i%12<6  :  j%12<6;
        fill(255 * int(aij));
        rect(10 * x, 10 * y, 10, 10);
        j++;
    }
    i++;
}
```

The program works with x and y coordinates which are multiplied by 10 to get black and white cells of 10×10 pixels. The variables i and j count warp threads and weft threads, respectively. We use a trick by initializing i at 1000. This guarantees i$-$j≥ 0. Therefore the % operator implements the mathematical *mod* correctly. The formula belongs to a family of formulas, producing successive Pied de poules. In Figure 3 we show four Pied de poules patterns, where the above formula generates the third pattern.

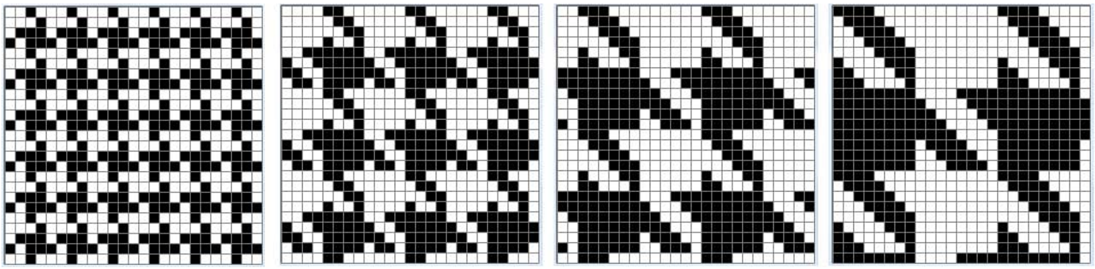


Figure 3 : *Successive Pied de poule patterns.*

The leftmost pattern of Figure 3 is sometimes called Pied de poule, but in Dutch it is often called “molenwiek” (the wings of a windmill). In English it is also called puppytooth. Later we shall come back to the status of the “molenwiek”, whether it is a true Pied de poule or not. The general formula is $(i-j)\%2N < N ?\ i\%4N < 2N :\ j\%4N < 2N$.

An alternative representation would be to use regular expressions e.g. $\{0^6 1^6\}^*$ for a weft pattern. It is at least as elegant and compact as $i\%12 < 6$, but the latter formula can go straight into the program (without requiring an interpreter). The weaving part is not easily done by a regular expression, it is two-dimensional.

What happens when N becomes larger and larger? The basic figures[1] in the Pied de poule become larger and larger (as shown in Figure 3), so eventually it becomes impractical to show them while keeping the cell size (weft thread width × warp thread width) fixed. But we can generate a Pied de poule for large N and then scale back by a scaling factor, S say, inversely proportional to N. This is how we made Figure 4 (left). Now we see that there is a limit Pied de poule. The size can be chosen arbitrarily, but the basic figure is fixed[2]. The *largest* Piede poule pattern formally does not exist, but there is a limit case obtained as $N \to \infty$.

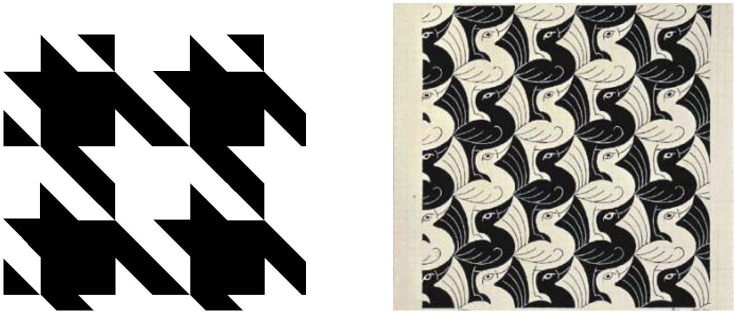


Figure 4 : *Limit case Pied de poule pattern and M.C. Escher, Birds E128.*

It is not practical to manufacture N/N woven Pied de poule for large N because the long thread loops make the fabric fragile. The patterns for large N, and the limit Pied de poule pattern *do* find application in fashion nevertheless, but more sophisticated weaving techniques must be used; often they are printed.

3 tessellations

The limit pattern falls apart in pieces: the black pieces touch each other pairwise in one point. If we cut them at this point, we can identify a single black piece, which we simply call the "pied". Inspired by this we can identify a pied for the patterns obtained for finite N, for example $N = 2,3,4$. These are shown in Figure 5 (after scaling so they are of comparable size). Actually, the space left between the black areas has the same shape; these are the white pieds. In this section we shall study in more detail how the individual pieds can be fitted together again to fill the entire two-dimensional plane. First we notice that the pied for finite N is not entirely symmetric. The limit case pied *is* symmetric, however. Reflecting the limit case pied along the indicated diagonal line yields the same figure again (Figure 5). The pieds become 'more symmetric' when N becomes larger.

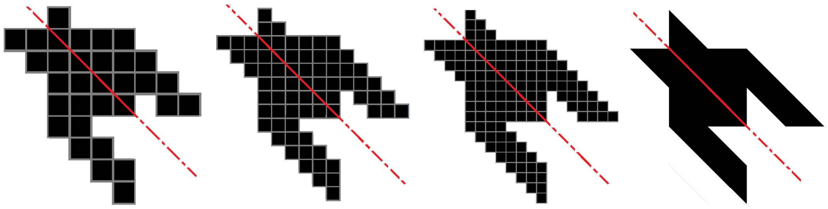


Figure 5 : *Successive pieds and limit pied.*

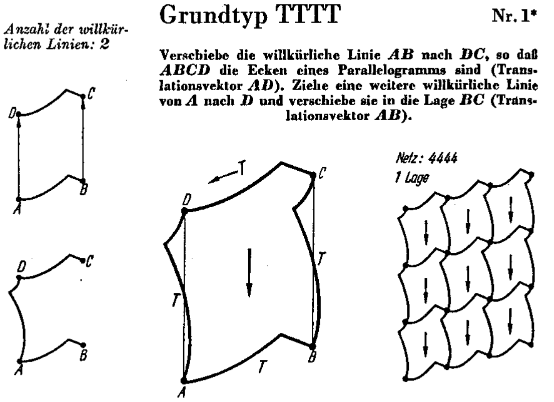
A tessellation (or tiling) is a collection of figures that fill the 2D plane with no overlaps They are used in textile design, interior design and industrial design [6]. M.C. Escher (1898-1972) added meaning to the tiles (e.g. animals), see Figure 4 (right) or [4, 5]. The mathematical theory of tessellation rests on two cornerstones: group theory, and topology. A geometric transformation which maps a figure to itself, is a symmetry of that figure. Certain transformations such as certain rotations, translations, reflections and

[1] Note that we have no formal definition of "basic figure" yet; actually the concept is still ambiguous as it depends on how we choose to cut, but after further formalization, we shall see it is the interior of a certain closed line, as for example in Figure 9.

[2] The limit also appears replacing integers i, j with reals x, y. As one reviewer noted, this Mathematica code makes the limit pattern: `RegionPlot[ If[Mod[(x - y), 2] < 1, Mod[x, 4] < 2, Mod[y, 4] < 2], {x, 0, 8}, {y, 0, 8}],`

combinations thereof, have been proven to be symmetries. A collection of operations including inverses (undoing the transformation) and compositions, is a *group*. In two dimensions one can create patterns filling the entire plane such that the pattern remains the same under certain transformations. For a specific pattern, this set of transformations forms a group, called a *wallpaper group*. There exist 17 wallpaper groups [7]. They appear in wallpaper, brickwork, floor tessellations, Eschers work, and indeed Pied de poule.

It is tempting to assume that each of the 17 regular wallpaper patterns gives rise to precisely one schema of regular tessellation. However, the relation is not one-to-one. What is still missing from the wallpaper patterns is: where are the cutting lines between the tiles? This question has been explored by Heesch and Kienzle and published in 1963 [8]. They added a topological analysis of the networks formed by the cutting lines between the tiles and gave 28 tessellation types, in a prescriptive style (in German). In Fig. 6 (left) and 7 (left) we show original descriptions of their first two tessellation types (the book is not widely available). The birds of Fig. 4 (right) follow the model Nr. 1. of Fig. 6 (left)[3].


Anzahl der willkürlichen Linien: 2

Grundtyp TTTT **Nr. 1***

Verschiebe die willkürliche Linie *AB* nach *DC*, so daß *ABCD* die Ecken eines Parallelogramms sind (Translationsvektor *AD*). Ziehe eine weitere willkürliche Linie von *A* nach *D* und verschiebe sie in die Lage *BC* (Translationsvektor *AB*).

Netz: 4444
1 Lage
D
C
A
B
D
C
A
B
T
D
C
T
T
T
A
B

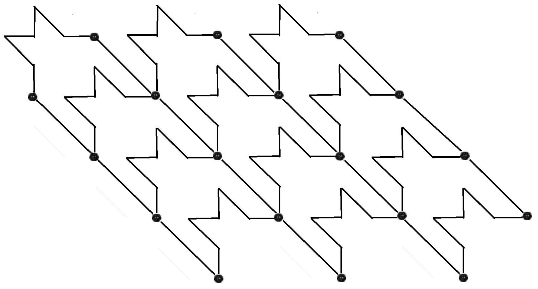


Figure 6 : *Prescriptive model and Limit case Pied de poules tessellation*

Which Heesch-Kienzle type describes the Pied de poule patterns? Compare the regular patterns of Figures 3 and 4 to the models of Figure 6 (left) and 7 (left). Type Nr. 1 is in concordance with the tessellation of the limit case Pied de poule of Figure 4 (left). The tessellation is shown explicitly by omitting the color and showing the cutting lines, in Figure 6 (right) – we added the network points manually.

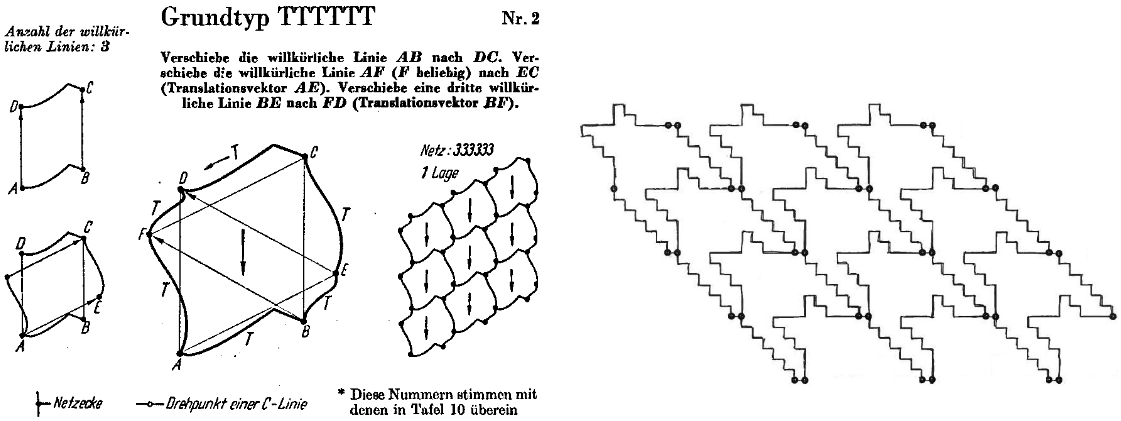
We expected that the patterns for finite N form the same type of tessellation, but there is a surprise. The successive patterns for $N = 2,3,4$ of Figure 3 can *not* be described by Nr. 1. The point where four black and four white pieds come together just is not there. Actually we need Heesch-Kienzle type Nr. 2. This is applicable for the first three pieds of Figure 5 and in fact for any $N > 1$. Also see Figure 7.

4 Drawing the contour

We noticed that the cutting lines form an alternative characterization of Pied de poule. So what are these contour lines? Can we program them? For drawing lines we felt Turtle Graphics commands would be a good candidate. `FD` abbreviates `FORWARD`, `RT` is `RIGHT` and `LT` is `LEFT`. So `RT 90` indicates turning clockwise over 90° and 1 is a distance. Asterisks are comments (interesting points).

```
LT 90 FD 1  * RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1
LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 FD 1
* FD 2 LT 90 FD 1 RT 90 FD 1 LT 90 FD 1  RT 90  FD 1 LT 90 FD 1 RT 90 FD 1
RT 90 FD 3 LT 90 FD 2 RT 90 FD 1 RT 90 FD 1 LT 90 FD 1 RT 90 FD 1 LT 90 FD 3
```

[3] Figs. 6 (left) and 7 (left) With kind permission from Springer Science+Business Media: Flächenschluss; System der Formen lückenlos aneinanderschliessender Flachteile. Heesch, H., & Kienzle, O., 1963, Figs. TTTT Nr1, p.64. and TTTTTT Nr2, p.64.



Grundtyp TTTTTT Nr. 2

Anzahl der willkürlichen Linien: 3

Verschiebe die willkürliche Linie *AB* nach *DC*. Verschiebe die willkürliche Linie *AF* (*F* beliebig) nach *EC* (Translationsvektor *AE*). Verschiebe eine dritte willkürliche Linie *BE* nach *FD* (Translationsvektor *BF*).

* Diese Nummern stimmen mit denen in Tafel 10 überein

Figure 7 : *Model Nr.2 of Heesch and Kienzle's theory and Pied de poules tesselated thus.*

Turtle Graphics have been used in children's mathematics education [10] and in 3D art [9]. In this way we get the line of Figure 8 (left), half the basic tile's contour. We added the 'interesting' points manually. The small triangle gives the starting position and orientation of the turtle. Using a Speedy300 we laser cut the contours generated by a Turtle Command interpreter for $N = 3$. The pied has become a puzzle piece. A few of the 50 pieces we cut are shown in Figure 8 (center).

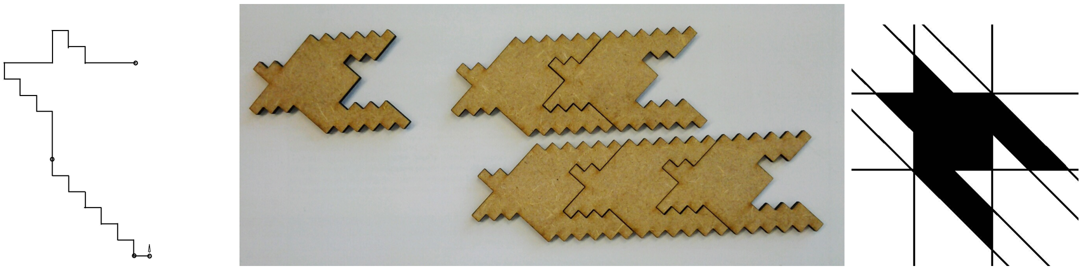


Figure 8 : *Half contour by turtle graphics (left), laser cut pieces (center), basic limit pied (right).*

One can split the program above into three parts (the '*' mark the splits). Then we can combine the command sequences for these parts with backward versions of the parts and thus draw the whole contour without lifting the pen. Yet the sequence of length 52 is unsatisfactory. The full contour is 104 (but from an information-theoretic viewpoint it is at most 52 plus a coding of the reversal and glue actions needed). But the list of 52 commands is neither compact, nor elegant, nor does it provide much insight. Re-arranging and subroutining the sequence is messy: Does a rotation belong to the next segment, or to the previous segment? Perhaps each rotation and one move could be packed together in a single command. We need a guiding principle. We found it by noting that the limit pied is be bounded by straight lines belonging to three sets: horizontal, vertical and one direction of diagonal[4]. See Figure 8 (right). So when a staircase has to be drawn, its procedure should be based on drawing one of these line types (adapted to the staircase situation).

We propose a variation of turtle graphics: we keep the idea of letting the turtle work relative to its own position, but the direction becomes absolute, rather than relative. It is a kind of sailor's language: think of a sailor at sea with a chip log (also called common log or ship log) and a compass, but no clock or sextant. Such a sailor would know relative positions and absolute directions. Obvious commands are `NORTH`, `SOUTH`, `EAST` and `WEST` (and their abbreviated version `N`, `E`, `S` and `W`). We call the command language Compass Logo. Now we program functions for going South-East or North-West in a staircase-wise manner. For each we have two versions: `ES` means going east first and then south whereas `SE` means going south first and then

[4] By algebra we find the lines replacing '<' by '=' in each component of $(i/S - j/S)\%2N < N\,?\,i/S\%4N < 2N : j/S\%4N < 2N$ scaling by $S = c/N$. The first is $(ciN - cjN)\%2N = N$, solutions $i = j{+}1/c$, $i = j{+}2/c, \ldots$ (slope -45^{o}, in Processing y goes down).

east. We show them in Figure 9. Function definitions and conditionals are as in Berkeley Logo (UCBL), www.cs.berkeley.edu/∼bh/logo.html.

```
TO ES :n
   IF :n = 0 [ STOP ]
   EAST 1
   SOUTH 1
   ES :n - 1
END
```

```
TO SE :n
   IF :n = 0 [ STOP ]
   SOUTH 1
   EAST 1
   SE :n - 1
END
```

```
N  2
ES 2
E  3
ES 6
W  3
NW 2
S  2
```

```
W  3
SE 3
S  3
WN 6
N  3
WN 3
E  3
```

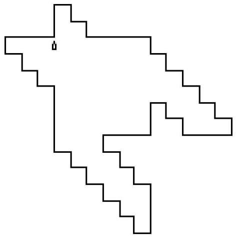


Figure 9 : *Compact Compass Logo program for contour.*

The same can be done for NW (North West) and WN (West North). The contour of the pied becomes straightforward, see the third and fourth box of Figure 9, whose output is in Figure 9 (right). We wrote an interpreter in Mathematica 8.0 as a functional program (in Scott-Strachy denotational semantics style). The interpreter is not efficient but usable. There is a general pattern NORTH $N-1$ ES $N-1$ E N ES $2N$ W N NW $N-1$ S $N-1$ W N SE N S N WN $2N$ NORTH N WN N E N where NORTH is the command, N the parameter.

Recall the windmill or puppytooth given by Figure 3 (left). The turtle graphics approach of Section 4 provides us with a tool to explore this case. Fill in $N = 1$ (and remove useless commands such as NORTH 0). Then we find automatically the Compass Logo command sequence for the contour of the minimal pied, shown in Figure 10 (left). When we ran the program for the first time we expected to see the windmill pattern (Figure 10 center), but to our great surprise we got the rightmost drawing of Figure 10 instead. This was not what we expected. Yet we found no error and we came to realize that this *is* the minimal pied

```
E  1
ES 2
W  1
W  1
SE 1
S  1
WN 2
N  1
WN 1
E  1
```

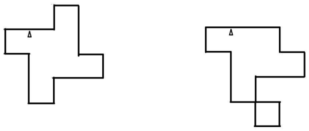


Figure 10 : *Program for contour, naive expectation (left) and true minimal Pied (right)*

Is the true minimal pied evidence that the term puppytooth is correct and is "windmill" wrong? First we must check for a third possibility: perhaps the windmills belong to another family of patterns as well? This is the case indeed (yet not explaining "windmill"). We found another family, viz. $a_{ij} = (i-j)\%2 < 1\ ?\ i\%4N < 2N\ :\ j\%4N < 2N$. For $N = 1$ it produces the same as Figure 3 (left) and for $N \geq 2$ the checkers shown in Figure 11 (called Pepita; used in chefs trousers and Porsche car interiors, amongst others). Again there is a limit pattern. In the limit we get squares areas with luminosity being the average of the areas with 50% black and 50% white squares: gray (it is unclear what the basic figure is). Figure 11 gives the family structure Venn-diagram. To our surpise, the "windmill" does not stand out as something common to the checkerboard family. The term "windmill" refers to a Gestalt which exists for $N = 1$ only.

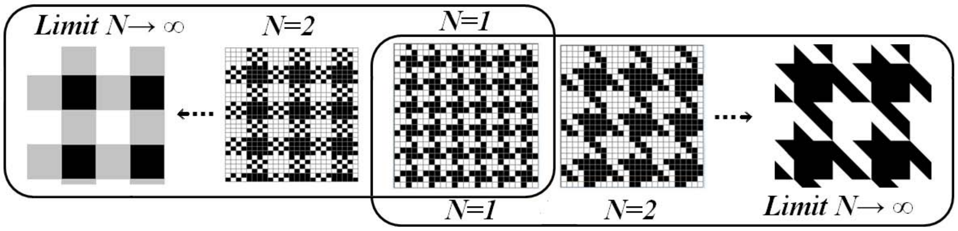
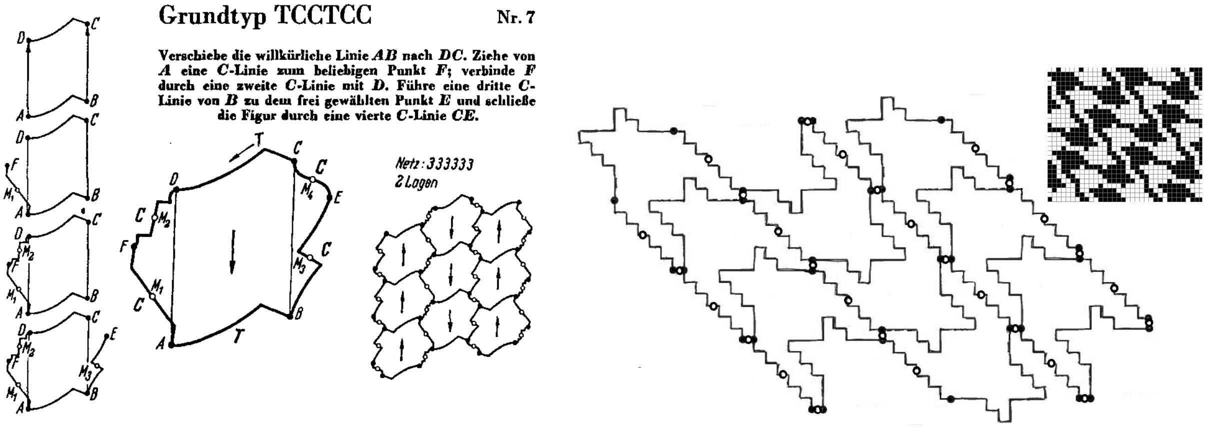


Figure 11 : *Venn diagram of Checkerboard and Pied poule patterns.*

5 New tessellations

Playing with the laser cut tiles we noticed that they interlock for $1 < N < \infty$, see Figure 8 (center). Once head-tails interlocked, they cannot be pulled apart without lifting. The tessellation has infinitely long staircase cutting lines; it falls apart in strips of interlocked pieds, which can be shifted with respect to each other.



Grundtyp TCCTCC Nr. 7

Verschiebe die willkürliche Linie *AB* nach *DC*. Ziehe von *A* eine *C*-Linie zum beliebigen Punkt *F*; verbinde *F* durch eine zweite *C*-Linie mit *D*. Führe eine dritte *C*-Linie von *B* zu dem frei gewählten Punkt *E* und schließe die Figur durch eine vierte *C*-Linie *CE*.

Figure 12: *Model Nr.7 with translations (T) and* 180° *rotations centers (C) and novel "pied de poule".*

First we guessed that there exists only one tessellation type (since the pied can be obtained by following the prescription Nr. 2). But playing with the strips more tessellations appear, based on translations along the infinite staircase cutting lines. In this way we find $2N - 1$ regular tessellations. We can also produce irregular tessellations (since the translations can be different at each line). We included rotations: rotate strips by 180° alternatingly. This fits because there are additional symmetries hidden in the sides of the basic pied which distinguish it from a Heesch-Kienzle tile which is really arbitrary ("willkürlich"). An additional tessellation is given in Figure 12 (right). The contour has line segments which are invariant with respect to 180° rotation around certain centers (the long staircase segments and the tail ends). This[5] is a Nr. 7. The tessellation of Figure 12 (right) is made of pieds for $N = 3$. The connection points and the rotation centers are indicated. This pattern is new to the best of our knowledge. The small rightmost figure shows a black and white pattern of this type for N=2. This does not exhaust the possibilities for creating novel patterns, such as adding reflection of entire strips. We leave this as an option for future research.

Acknowledgements: I like to thank Christoph Bartneck, Jun Hu, and Tom Verhoeff for the cooperation in creative programming, Katinka Feijs for the fashion inspiration, and the reviewers for valuable feedback.

References

[1] Gerdes, P. African Basketry: Interweaving Art and Mathematics in Mozambique, Proc. Bridges (Coimbra) 2011.
[2] Gerdes, P. African Basketry: A Gallery of Twill-Plaited Designs and Patterns Lulu.com, (2008).
[3] Kolmogorov, A. (1968). "Logical basis for information theory and probability theory". IEEE Transactions on Information Theory 14 (5): 662 -664.
[4] M.C. Escher, the graphic work. Taschen 2001.
[5] Doris Schattschneider. M.C. Escher: Visions of Symmetry. W. H. Freeman (1992).
[6] Feijs,L. and Bartneck,C. (2009) Teaching Geom. Principles to Design Students. Dig. Cult.&Educ., 1(2), 104-115.
[7] Fejes Tóth, L. (1964). Regular figures. New York,: Macmillan.
[8] Heesch, H. and Kienzle, O. (1963). Flächenschluss; System der Formen lückenlos aneinanderschliessender Flachteile. Berlin,: Springer.
[9] Verhoeff, T. 3D Turtle Geometry: Artwork, Theory, Program Equivalence and Symmetry. Int. J. of Arts and Technology, 3(2/3):288319 (2010).
[10] Seymour Papert. Mindstorms: children, computers, and powerful ideas. 2nd edition, 1993, Basic Books.

[5] Fig. 12 (left) With kind permission from Springer Science+Business Media: Flächenschluss; System der Formen lückenlos aneinanderschliessender Flachteile. Heesch, H., & Kienzle, O., 1963, Fig. TCCTCC Nr 7, p.67.