Functional Geometry Description of Escher's Fish

Milton Mazzarri

Feb 1, 2017

Houston Elixir Meetup

Introduction

Square Limit

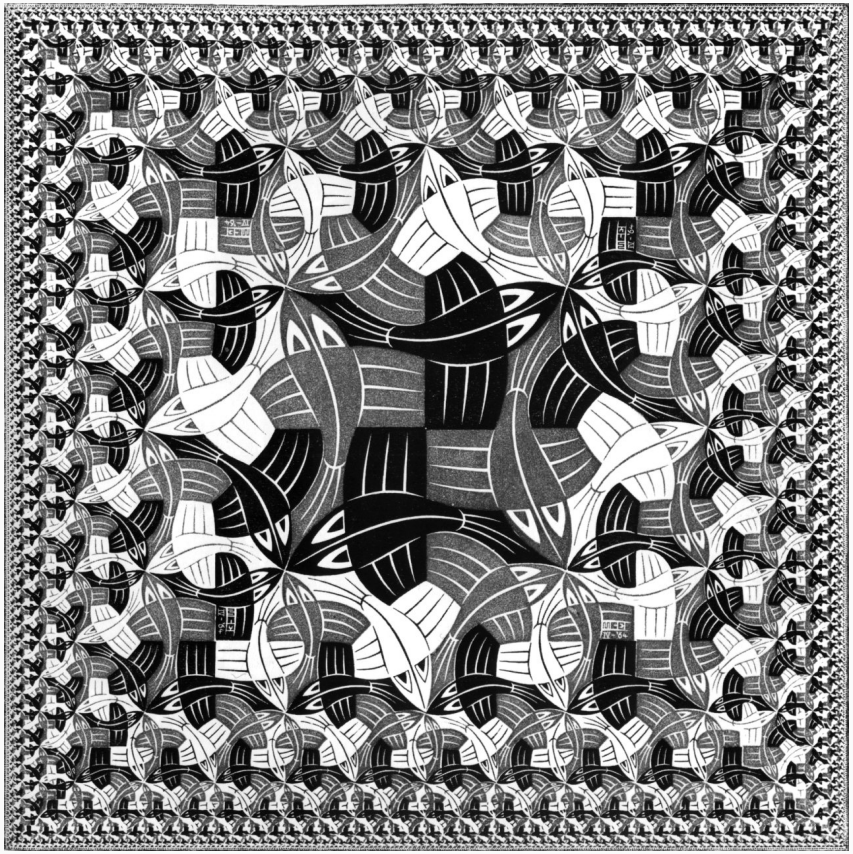

Figure 1: M.C. Escher's Square Limit

Source: `https://www.wikiart.org/en/m-c-escher/square-limit`

Functional Geometry

Functional Geometry is a paper by Peter Henderson[1, 2], which deconstructs the M.C. Escher woodcut "Square Limit".

> *A picture is an example of a complex object that can be described in terms of its parts. Yet a picture needs to be rendered on a printer or a screen by a device that expects to be given a sequence of commands. Programming that sequence of commands directly is much harder than having an application generate the commands automatically from the simpler, denotational description.*

Basic Operations

Note

The image it is located within a frame, but we do not consider the frame to be part of the picture.

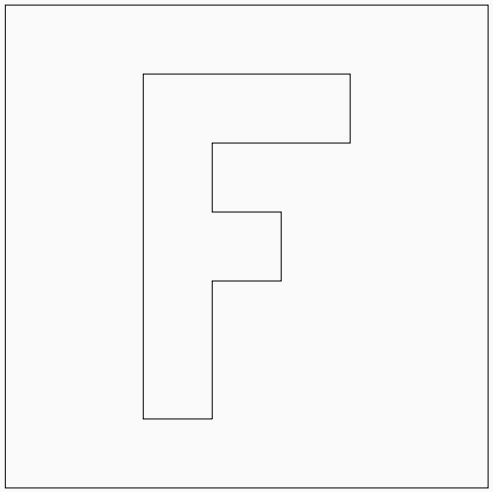

Figure 2: The value `f` denotes the picture of the letter F

Basic operations on pictures

- *rot*(*picture*) :: *picture*
- *flip*(*picture*) :: *picture*
- *rot*45(*picture*) :: *picture*
- *above*(*picture*, *picture*) :: *picture*
- *beside*(*picture*, *picture*) :: *picture*
- *over*(*picture*, *picture*) :: *picture*

Rotation

Rotates a picture 90° anticlockwise.

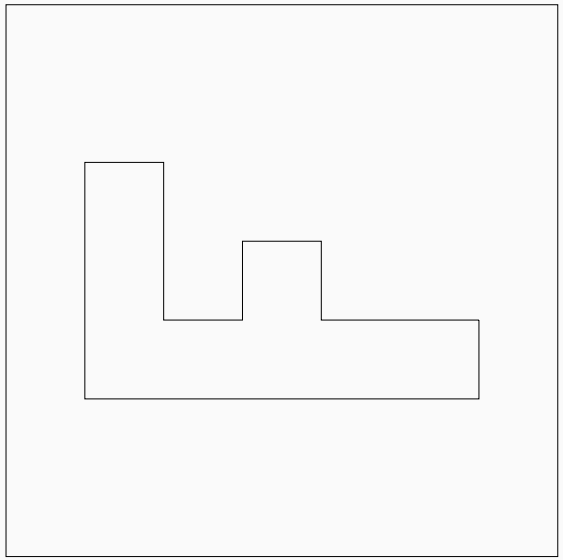

Figure 3: `rot(f)`

Flip

Flip a picture through its vertical centre axis.

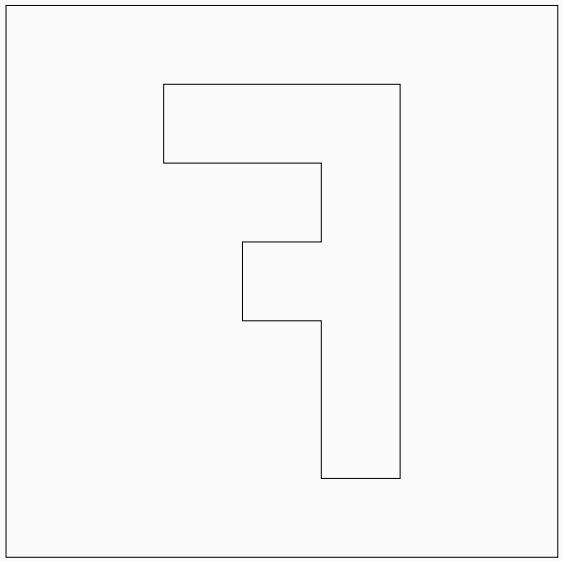

Figure 4: `flip(f)`

Rotation and Flip

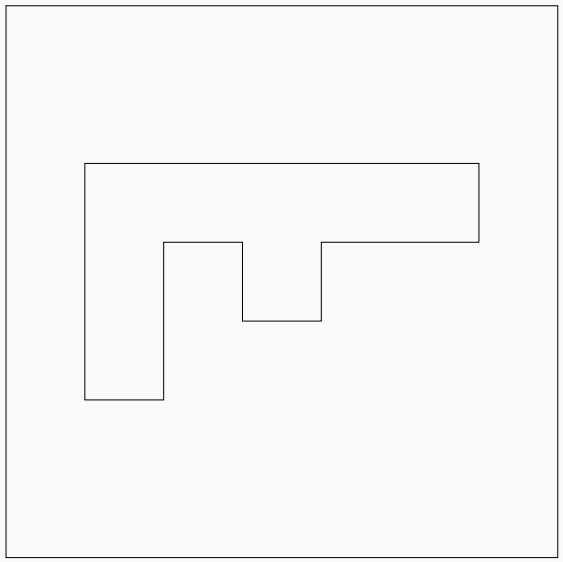

Figure 5: `rot(flip(f))`

Rotation 45°

Rotates a picture about its top left corner, through 45° anticlockwise.

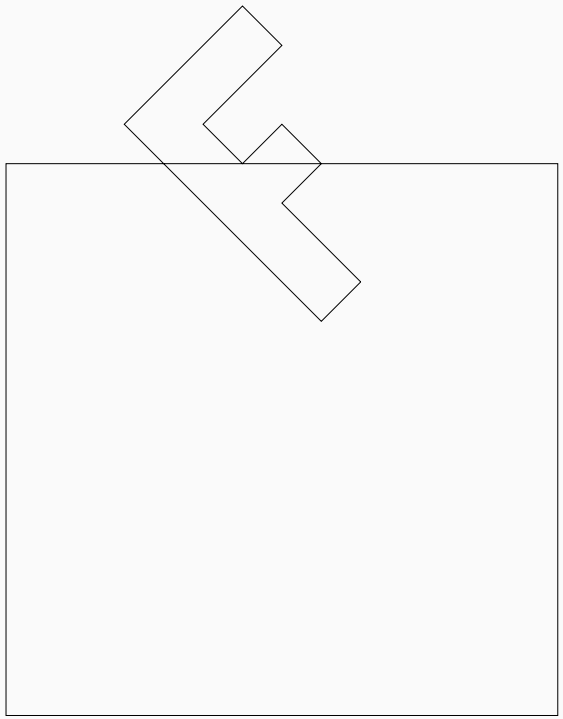

Figure 6: `rot45(f)`

Above

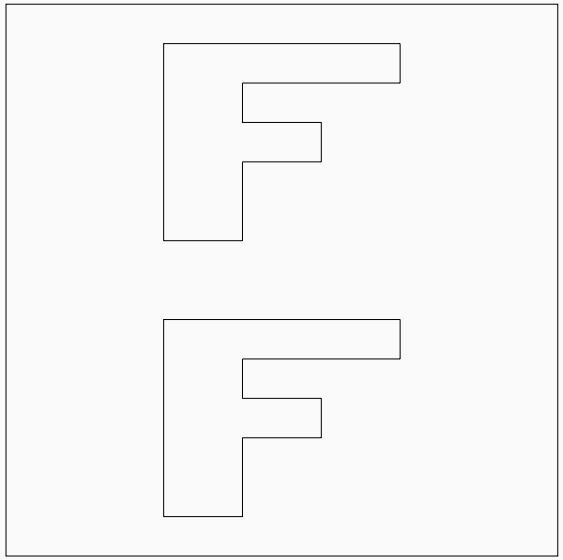

Figure 7: `above(f, f)`

Beside

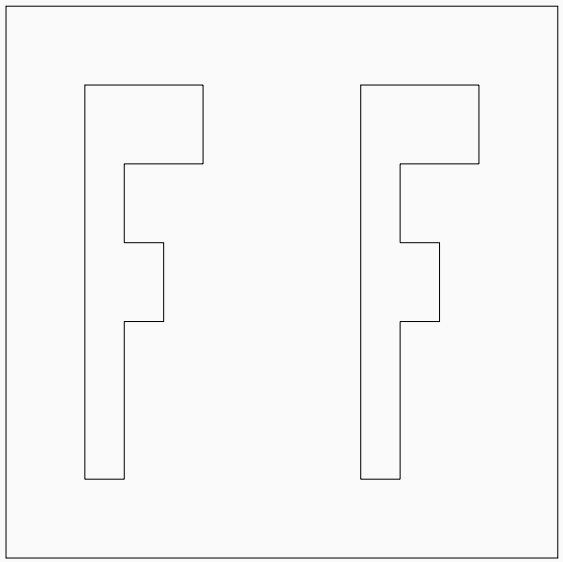

Figure 8: `beside(f, f)`

above/beside combination

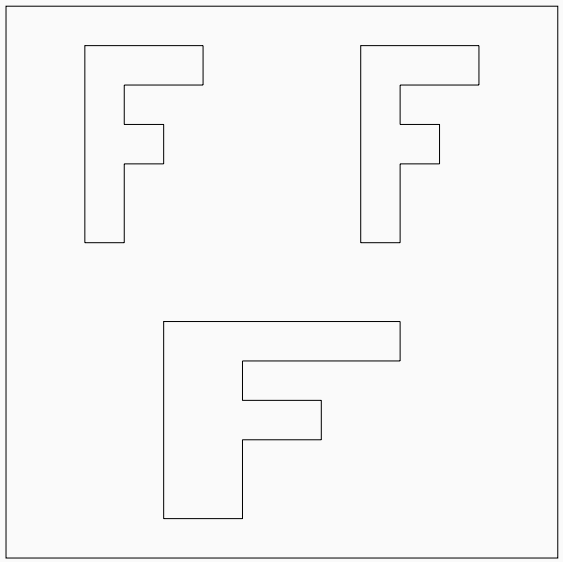

Figure 9: `above(beside(f, f) f)`

Superposition

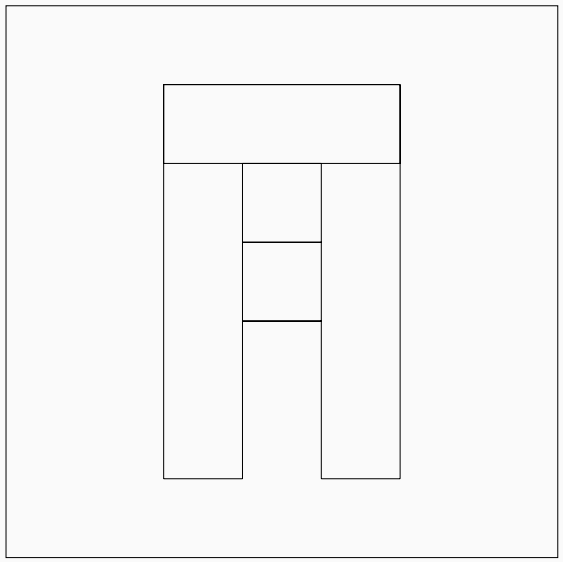

Figure 10: `over(f, flip(f))`

Laws

Laws (Unit Test)

$$rot(rot(rot(rot(p)))) = p$$

$$rot(above(p, q)) = beside(rot(p), rot(q))$$

$$rot(beside(p, q)) = above(rot(q), rot(p))$$

$$flip(beside(p, q)) = beside(flip(q), flip(p))$$

Square Limit

Basic patterns: p, q

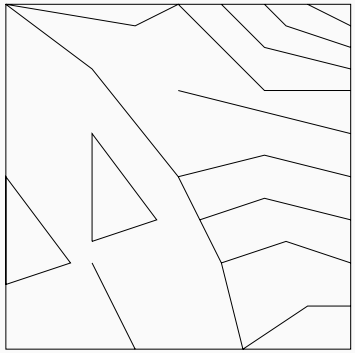

Figure 11: `p`

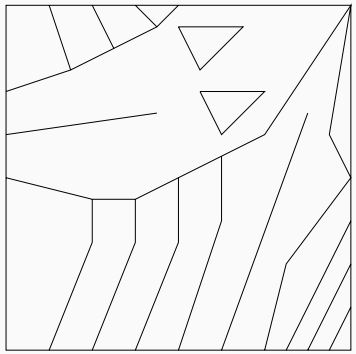

Figure 12: `q`

Basic patterns: r, s

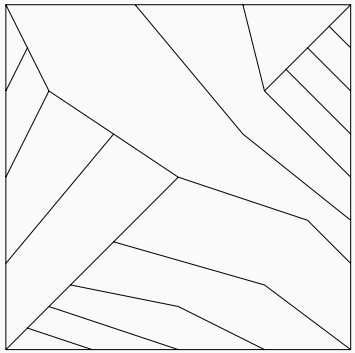

Figure 13: `r`

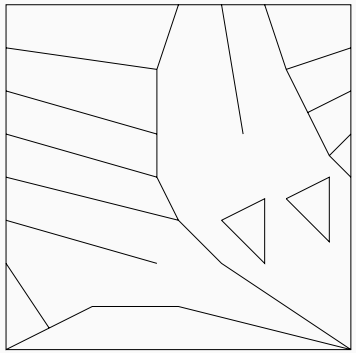

Figure 14: `s`

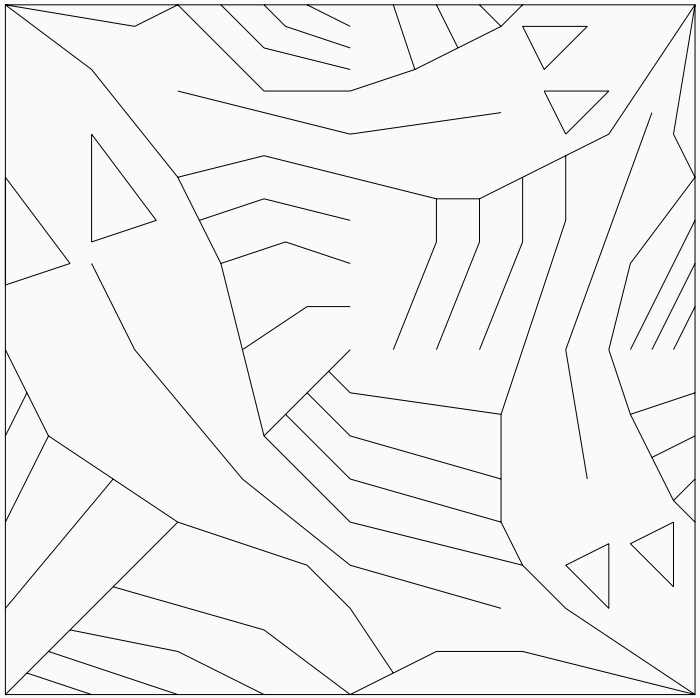

Figure 15: `t = quartet(p, q, r, s)`

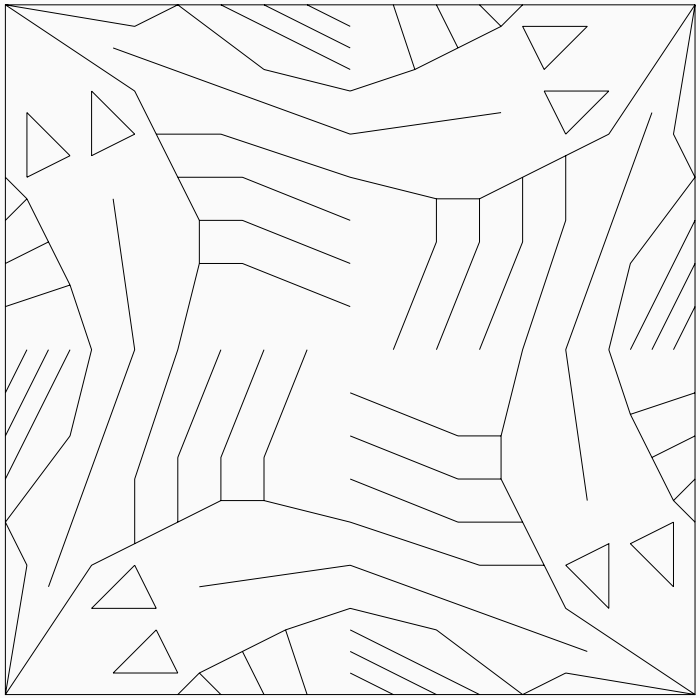

Figure 16: `u = cycle(rot(q))`

side1

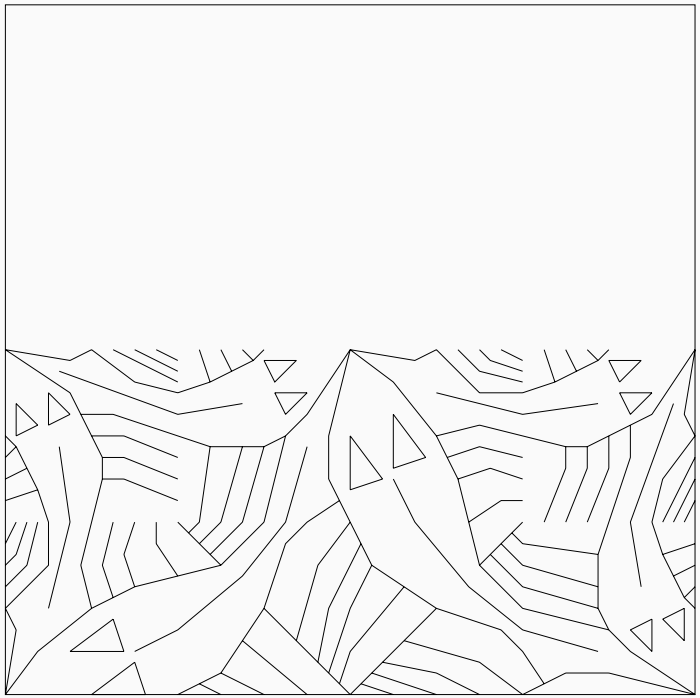

Figure 17: `side1 = quartet(blank, blank, rot(t), t)`

side2

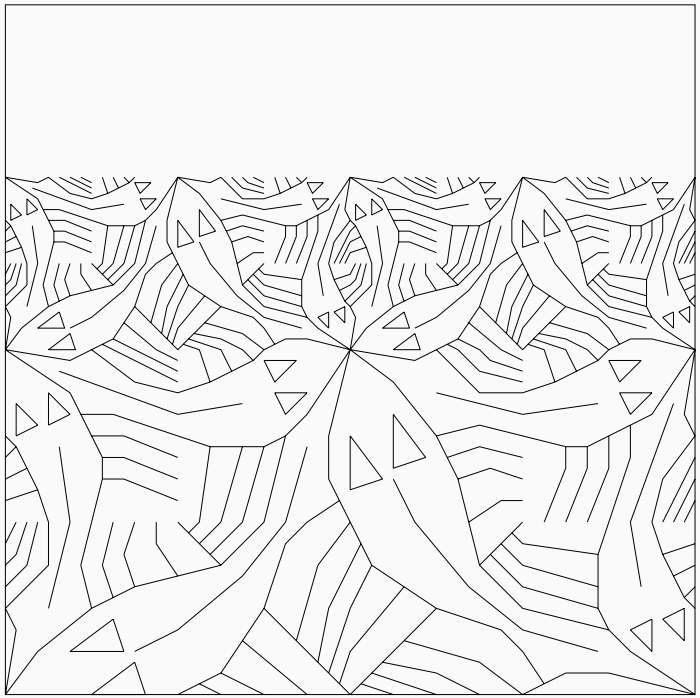

Figure 18: `side2 = quartet(side1, side1, rot(t), t)`

corner1

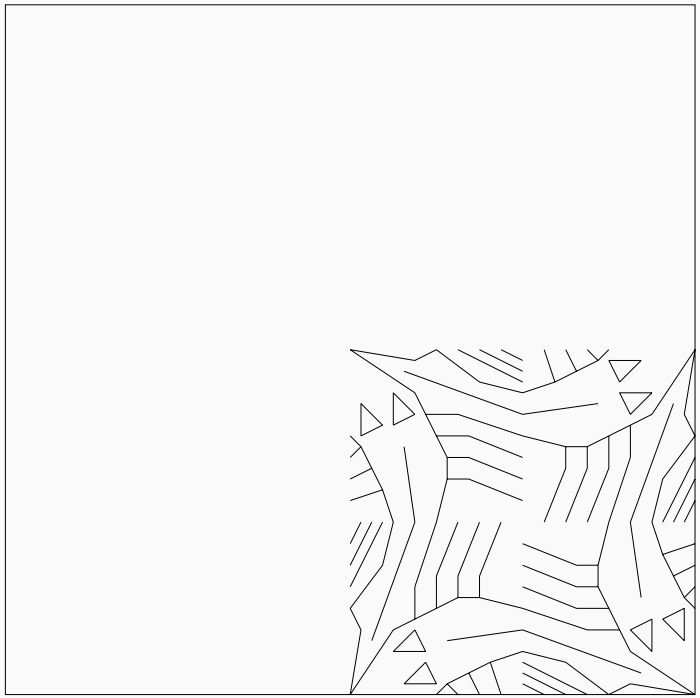

Figure 19: `corner1 = quartet(blank, blank, blank, u)`

corner2

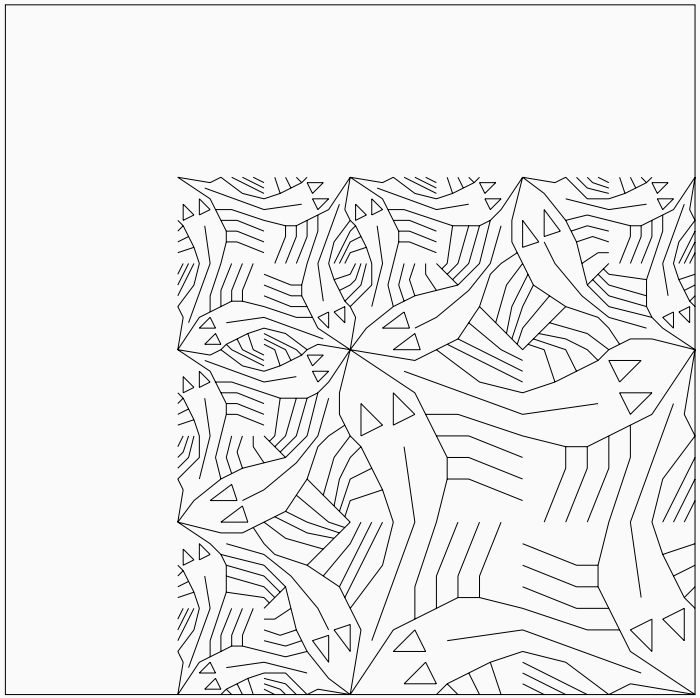

Figure 20: `corner2 = quartet(corner1, side1, rot(side1), u)`

pseudocorner

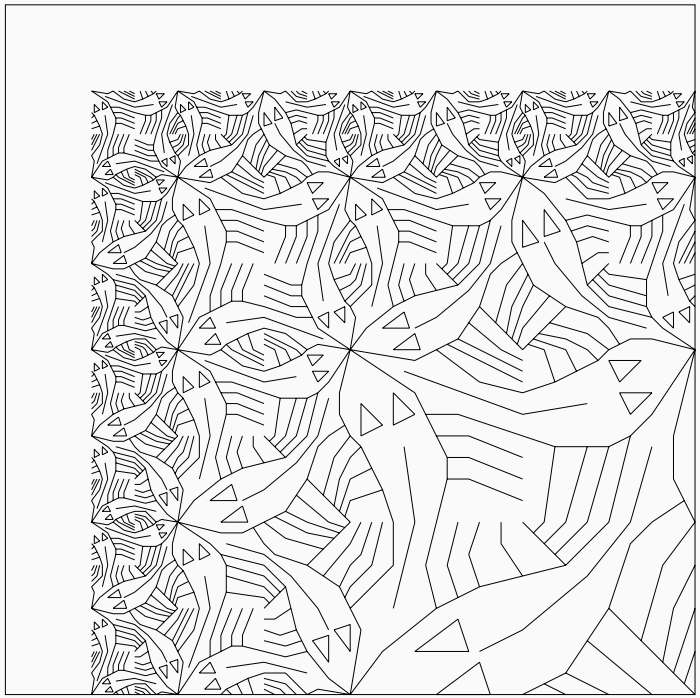

Figure 21: `pseudocorner = quartet(corner2, side2, rot(side2), rot(t))`

pseudolimit

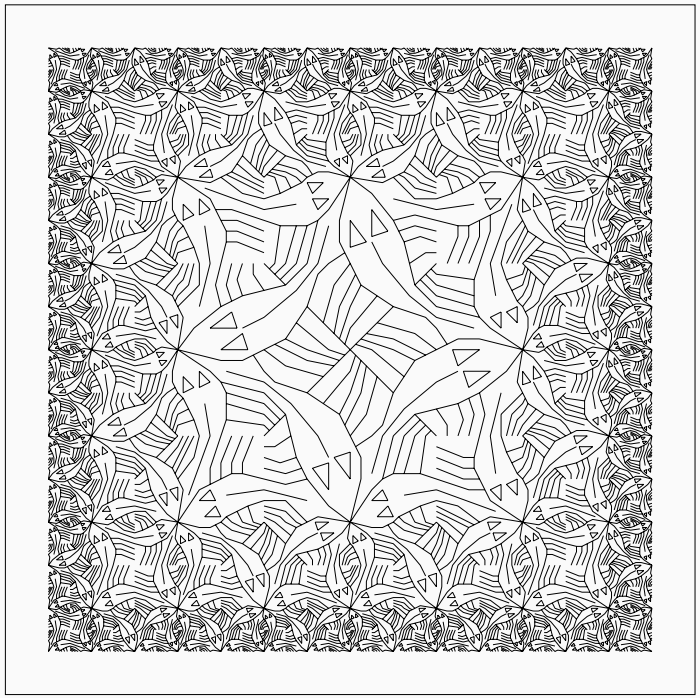

Figure 22: `pseudolimit = cycle(pseudocorner)`

corner

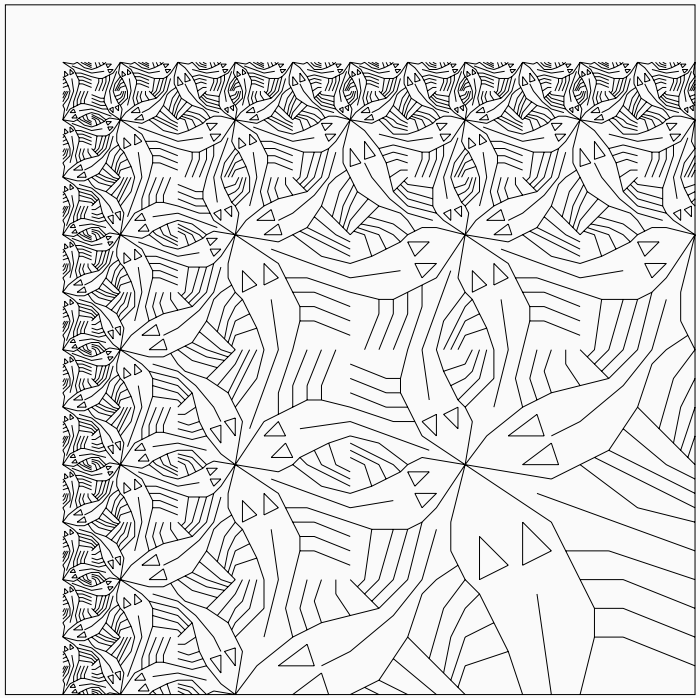

Figure 23: corner

squarelimit

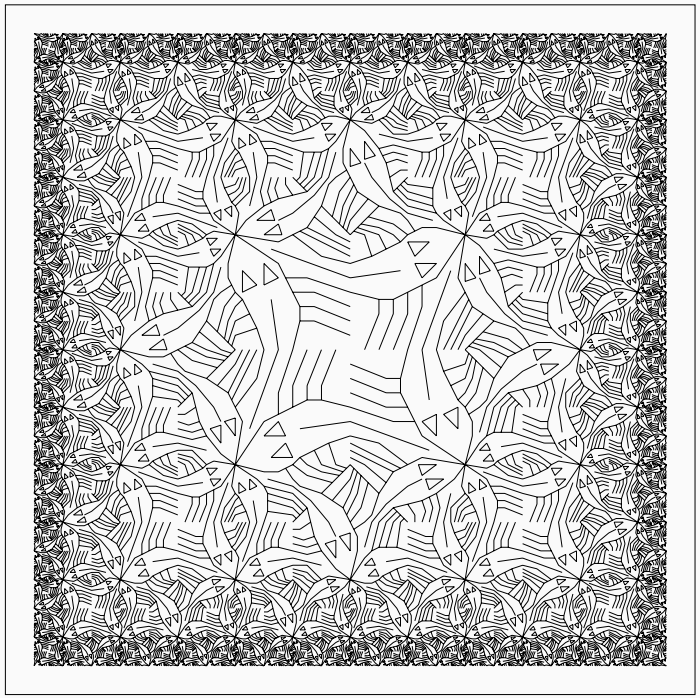

Figure 24: squarelimit

Implementation

Vectors

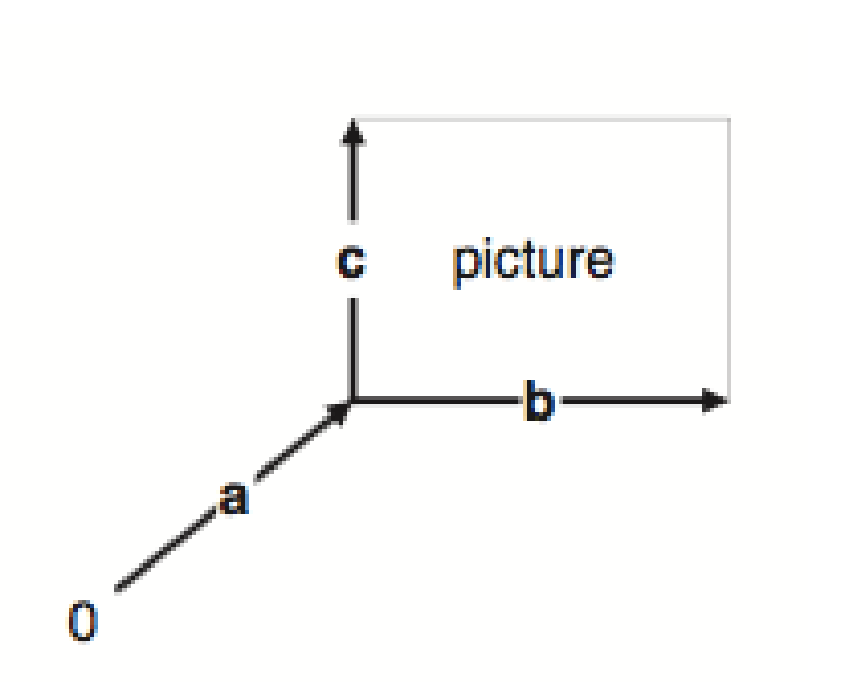

Figure 25: Basic vectors

Implementation

$$over(p, q)(a, b, c) = p(a, b, c) \cup q(a, b, c)$$

$$blank(a, b, c) = \{\}$$

$$beside(p, q)(a, b, c) = p(a, \frac{b}{2}, c) \cup q(a + \frac{b}{2}, \frac{b}{2}, c)$$

$$above(p, q)(a, b, c) = p(a, b, \frac{c}{2}) \cup q(a + \frac{c}{2}, b, \frac{c}{2})$$

Implementation

$$rot(p)(a, b, c) = p(a + b, c, -b)$$

$$flip(p)(a, b, c) = p(a + b, -b, c)$$

$$rot45(p)(a, b, c) = p(a + \frac{b + c}{2}, \frac{b + c}{2}, \frac{c - b}{2})$$

Demo

Future

- Add support for SVG Path.
- Escher's "Circuit Limit III" picture.

Circuit Limit III

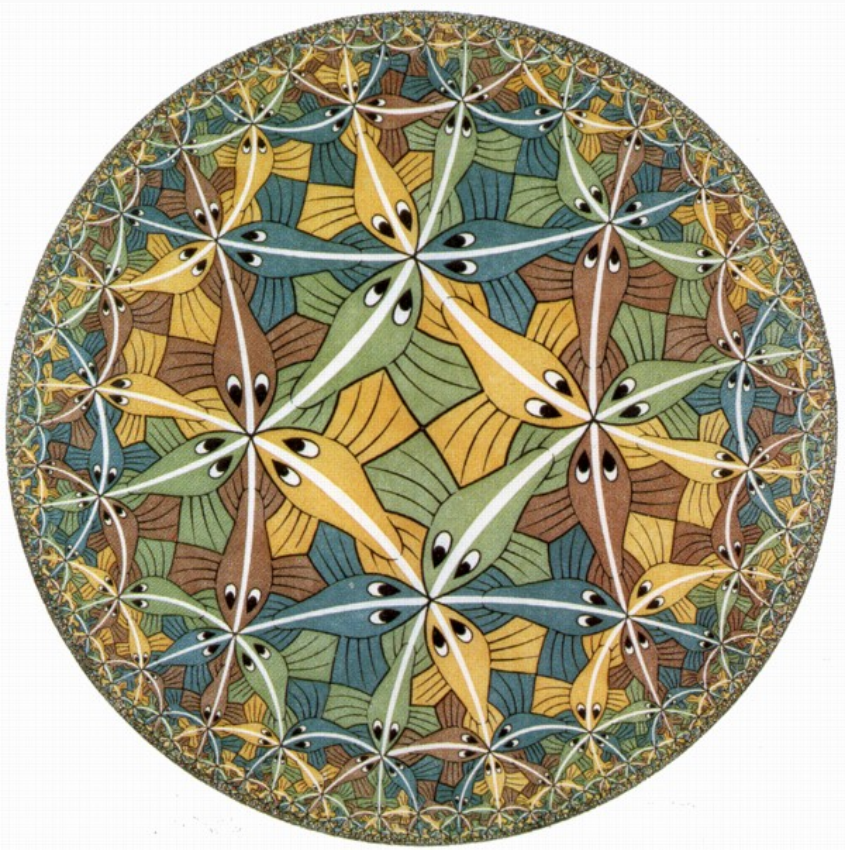

Figure 26: Circuit Limit III

Source: `http://mathstat.slu.edu/escher/upload/9/90/Circle-Limit-III.jpg`

Thanks!

`https://github.com/milmazz/func_geo`

References I

P. Henderson.

Functional geometry, 1982.

P. Henderson.

Functional geometry, 2002.