

Opt Art: Special Cases

Robert Bosch
Dept. of Mathematics
Oberlin College
Oberlin, OH 44074
(bobb@cs.oberlin.edu)

Abstract

We present special cases of edge-matched mosaics, TSP Art, and map-colored mosaics that do not require integer programming. They can be solved very quickly.

1 Introduction

At recent Bridges conferences, we have presented various examples of what we refer to as Opt Art: visual art constructed with the assistance of mathematical optimization techniques [1]. We have devised an integer programming (IP) formulation for designing edge-matched mosaics [2]. We have modified Dantzig, Fulkerson, and Johnson’s IP formulation for the Traveling Salesman Problem (TSP) so that it can be used to create simple closed curves that divide the plane into two regions that, when colored differently from one another, resemble knots and links [3,4]. And we have introduced two IP formulations for designing map-colored mosaics [6].

Integer programming is a widely used tool for tackling difficult combinatorial optimization problems like the Traveling Salesman Problem [11]. The most commonly used algorithm for solving IP problems is a form of divide and conquer known as branch and bound. The algorithm works by constructing a binary tree of linear programming problems. It solves the linear programming problems with the Simplex Method, which is fairly fast in practice (even though it requires an exponential number of iterations in the worst case). Branch and bound tends to work well when the formulation is “tight” (when the optimal value of the linear programming relaxation is close to that of the integer programming problem). Fortunately, in most opt-art instances, the formulations are tight, so the number of nodes in the tree (the number of linear programming problems that end up being solved) tends to be small. The drawback is that the linear programming problems are fairly large, so the total amount of computation still ends up being nontrivial.

In the present paper, we introduce special cases of edge-matched mosaics, TSP Art, and map-colored mosaics that do not require integer programming. They can be solved very quickly.

2 Edge-matched mosaics

Mosaics are made out of tesserae (tiles). In an *edge-matched mosaic*, the tesserae are decorated geometric tiles. The decorations on the tiles make some tesserae brighter than others, and they form patterns on their edges. Edge-matched mosaics are characterized by these edge patterns. When building an edge-matched mosaic, the artist must lay down the tesserae so that the edge patterns on adjacent edges of adjacent tesserae match. Figure 1 displays an example based on a small section of Leonardo da Vinci’s *Mona Lisa* (her eyes).

We assembled this mosaic out of the set of tesserae displayed in Figure 2. We produced these shaded Truchet-like tiles using two circular arcs to divide a square into three pieces. We centered the arcs at opposite corners of the square, and we set the radii to one half the square’s side length. When we colored the pieces of the square, we allowed ourselves three colors—black, gray, and white—and we forced ourselves to give

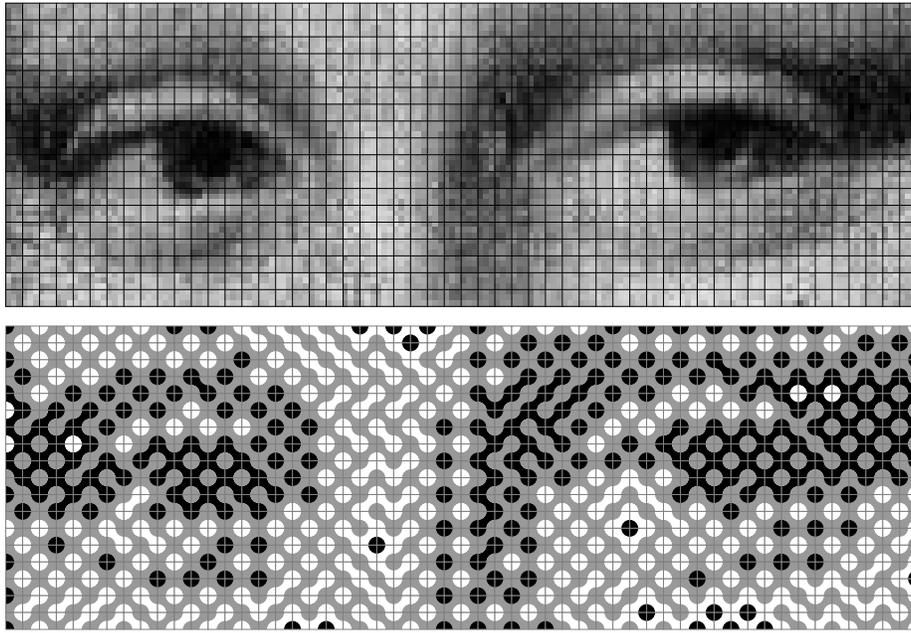


Figure 1: The eyes of *Mona Lisa* (top) rendered as an edge-matched mosaic (bottom).

neighboring pieces different colors. This system of decoration gave rise to six different horizontal edge patterns, $\{h_1, \dots, h_6\}$, and six different vertical edge patterns, $\{v_1, \dots, v_6\}$.

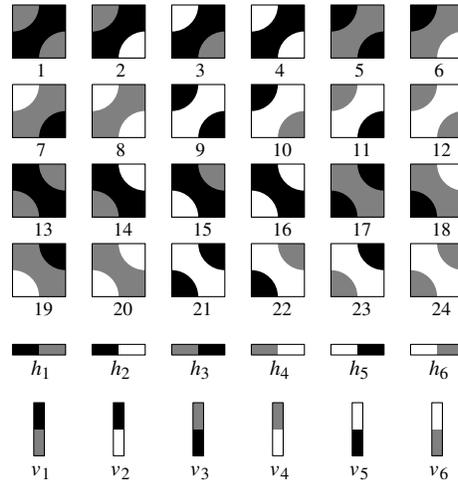


Figure 2: The tesserae (top) together with their horizontal (middle) and vertical (bottom) edge patterns.

To design the mosaic—to place the tesserae so that their edges match and they collectively resemble the eyes of *Mona Lisa*—we used an IP formulation similar to those described in [2,6]. Solving the integer program required more than an hour of CPU time on a Pentium IV laptop.

2.1 Edge-matched mosaics: a special case

It may come as a surprise that if we limit ourselves to the black-and-white tesserae (tesserae 4, 9, 16, and 21), we gain the ability to create high quality edge-matched mosaics very quickly. Tesserae 4 and 16 are dark,

each having an average brightness of $\pi/8 \approx 0.39$, and tesserae 9 and 21 are light, each having an average brightness of $1 - \pi/8 \approx 0.61$. For each tessera, precisely two of the others can be a neighbor (i.e., placed next to it on its right side, left side, above it, or below it), and one of these possible neighbors is dark and the other is light. (For tessera 4, the dark possible neighbor is 16 and the light possible neighbor is 9.)

We let our target image I be a $w \times h$ grid of grayscale values $I_{i,j} \in [0, 1)$, where 0 stands for black and 1 stands for white. We choose an integer $k \geq 1$, and divide the image into a grid of $\lfloor w/k \rfloor \times \lfloor h/k \rfloor$ square cells k on a side. We then let $A_{i,j}$ be the mean grayscale value of cell (i, j) . As each $A_{i,j} \in [0, 1)$, we can interpret the array of $A_{i,j}$ values as an image A , a “down sampled” version of I .

Next, we use Floyd-Steinberg dithering [8] to convert A into a grid of 0s and 1s, a binary (black-and-white duotone) image B . To construct the edge-matched mosaic, we start in the upper left corner. If the upper left entry of B is a 0, we can use either a copy of tessera 4 or a copy of tessera 16 (one of the dark tesserae). If it is a 1, we can use either a copy of tessera 9 or or a copy of tessera 21 (one of the light ones). And once we select the tessera for the upper left corner, the rest of the mosaic is completely determined. Every 0 will be covered by a copy of tessera 4 or of 16, and every 1 will be covered by a copy of tessera 9 or of 21. So in a sense, no errors will be made, as each 0 will get a dark tessera, and each 1 will get a light tessera. (Of course, the actual error in representing image B with the tesserae will be exactly $\pi/8$ times the total number of entries.)

The end result for the *Mona Lisa* eyes image is displayed in Figure 3 (top). It can be described as an edge-matched mosaic made out of black-and-white, duotone, Truchet-like tiles. A larger version is displayed in Figure 10 (bottom). (Truchet-like tiles were first studied by Pickover [10]. Browne recently discussed mosaicking with duotone Truchet-like tiles, but apparently did not realize that they can be used to render arbitrary source images [7].) The bottom portion of Figure 3 is an edge-matched mosaic based on an alternate set of tesserae that interact with each other in the exact same way that tesserae 4, 9, 16, and 21 do.

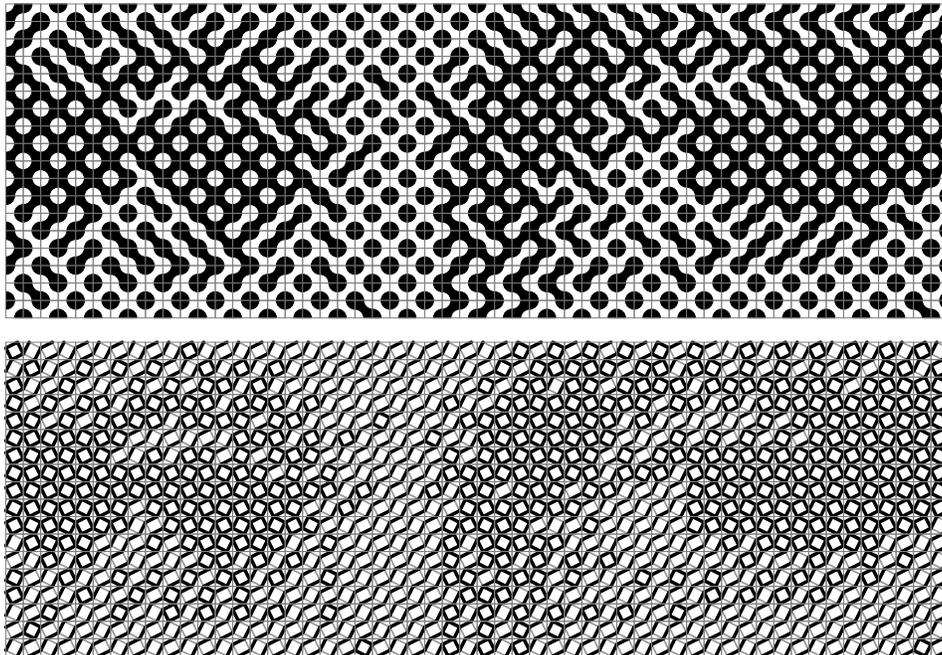


Figure 3: The eyes of *Mona Lisa* rendered as duotone edge-matched mosaics.

3 TSP Art

TSP Art was invented by Bosch and Herman [5] and greatly improved by Kaplan and Bosch [8]. To make a piece of TSP Art, we follow a two-step procedure. On the first step, we convert a source image into a stipple drawing, a collection of points. Once we’ve done this, we think of the points as an instance of the TSP, in which a salesman based in one city must visit each of the other cities exactly once before returning home, while minimizing total distance traveled. On the second step, we solve the TSP and draw the salesman’s optimal tour. From a computational standpoint, the first step (rendering the source image as a collection of points) is generally much easier than the second step (solving the TSP).

3.1 TSP Art: a special case

If our collection of points forms a grid graph, then the resulting TSP instance can be very easy to solve, often just by inspection. In a grid-graph instance, any tour made entirely of least-length edges (“minis”) is optimal.

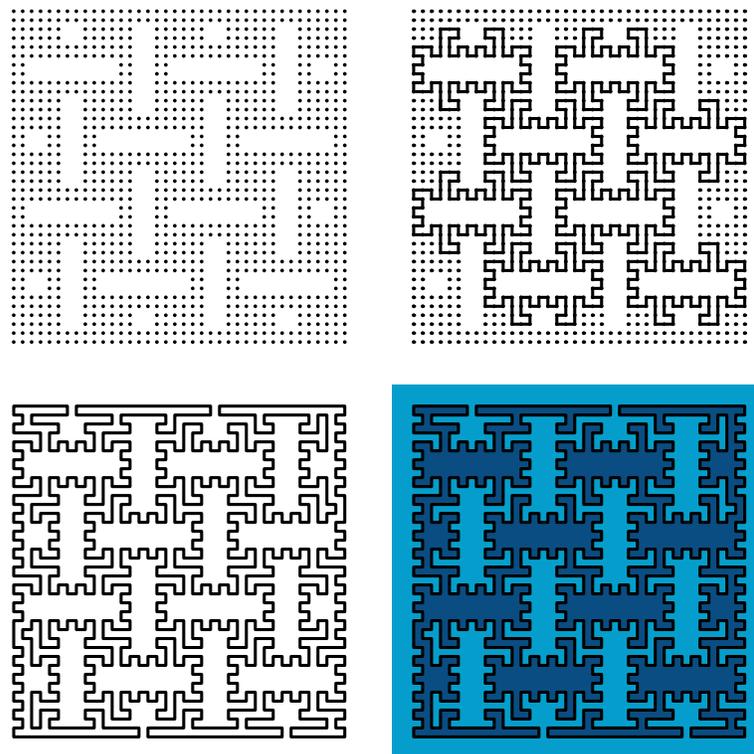


Figure 4: (a) a grid-graph instance of the TSP, (b) tiles whose interiors resemble pieces of horizontal strands, (c) an optimal tour, (d) with the interior and exterior colored differently.

The TSP instance displayed in Figure 4(a) has 1092 cities that collectively suggest a placemat woven from eight strands of material—four vertical, and four horizontal. This instance has many optimal solutions. To find the optimal solution displayed in Figures 4(c) and 4(d), we created a tile whose interior resembles one of the pieces of a horizontal strand. We designed this tile in such a way that when multiple copies are placed on the cities, as in Figure 4(b), the exterior of the tiles resembles the pieces of the vertical strands. We found the optimal tour by merging the tiles and adding additional edges, keeping in mind our goal of finding an optimal solution that has 180-degree rotational symmetry.

The TSP instance displayed in Figure 5(a) has 422 cities that collectively suggest a 4×4 grid of squares. To find the optimal solution displayed in Figures 5(b) and 5(c), we followed the same procedure described above (laying down tiles and then merging them). Here, our goal was to find a tour that divided the plane into two regions that, when colored, resemble a 4×4 checkerboard.

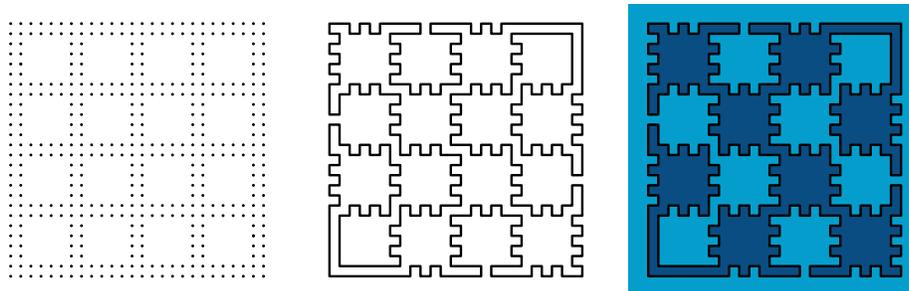


Figure 5: (a) another grid-graph instance of the TSP, (b) an optimal tour, (c) with the interior and exterior colored differently.

4 Map-colored mosaics

In a *map-colored mosaic*, the artist thinks of the mosaic as a map, its tesserae as countries, and adjacent tesserae (two that share an entire edge) as neighboring countries. In order for a map to be properly colored, neighboring countries must be painted with different colors. So when building a map-colored mosaic, the artist must lay down the tesserae so that adjacent tesserae are differently colored. If the number of available colors is large, this is easy—if the map-coloring constraint prevents the artist from using the desired color, he or she one can use a different color that's close to the desired color. But if the number of available colors is small, it can be quite difficult.

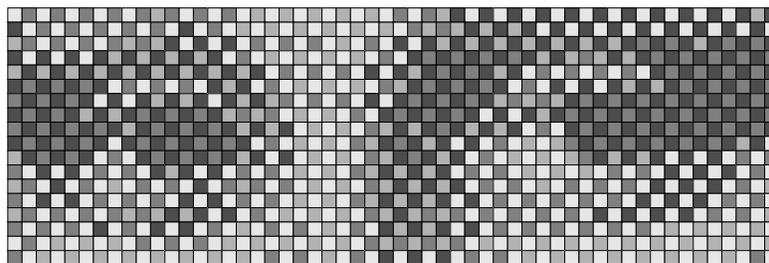


Figure 6: The eyes of *Mona Lisa* rendered as a map-colored mosaic.

Figure 6 displays an example based on the *Mona Lisa* eyes image. Only four colors (four different shades of gray) appear. The design was constructed with an IP formulation that required several hours of CPU time to solve.

4.1 Map-colored mosaics: a special case

Surprisingly, one can construct very high quality map-colored mosaics that use only two colors. One possibility is to overlay the image with a two-colorable tiling, as in Figure 7.

Here each square cell of the source image matches up with a pair of triangles of the tiling. When we color the tiling black and white, one of the triangles will be black and the other will be white. So the square formed by the two triangles will have an average brightness of 0.5. Consequently, if we take the edge shared

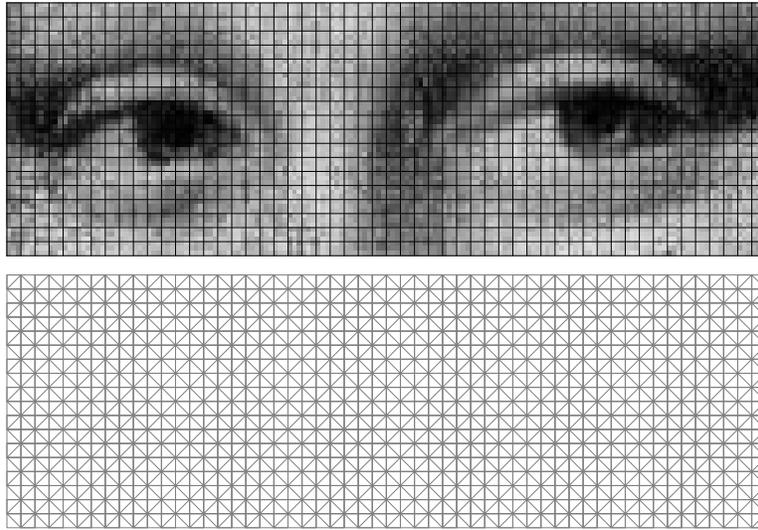


Figure 7: The eyes of *Mona Lisa* (top), and a two-colorable tiling (bottom).

by the triangles and bend it, we can adjust the average brightness of the square. If we bend the edge so that it extends into the black triangle, the square will become brighter. If we bend it so that it extends into the white triangle, the square will become darker. The idea is to make the bending of an edge based on the brightness of the corresponding square in the source image. As long as we restrain ourselves when we do our edge bending, the new tiling we obtain will remain two-colorable. And the colored version will resemble the source image quite closely. Figure 8 illustrates the initial (top) and final (bottom) colored tilings.

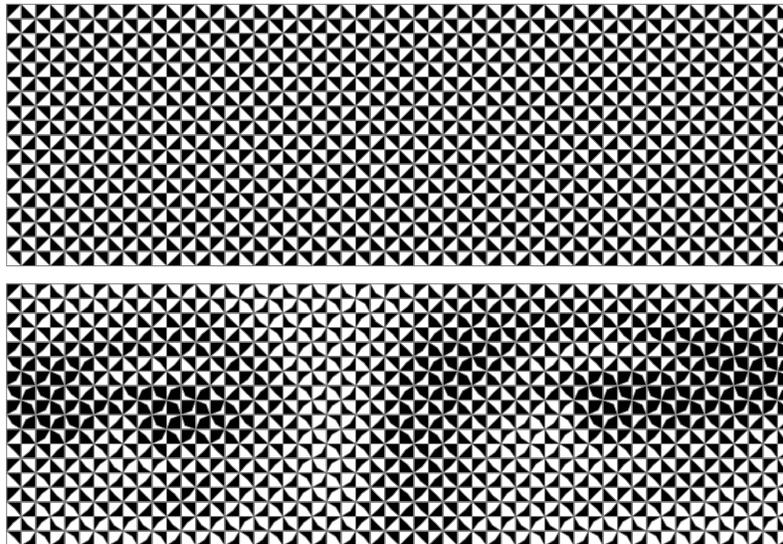


Figure 8: Initial colored tiling (top), and the final version (bottom).

Many variations of this process are possible. Two are displayed in Figure 9. In the bottom one, all of the tiles are convex. A larger mosaic is displayed in Figure 10 (top).

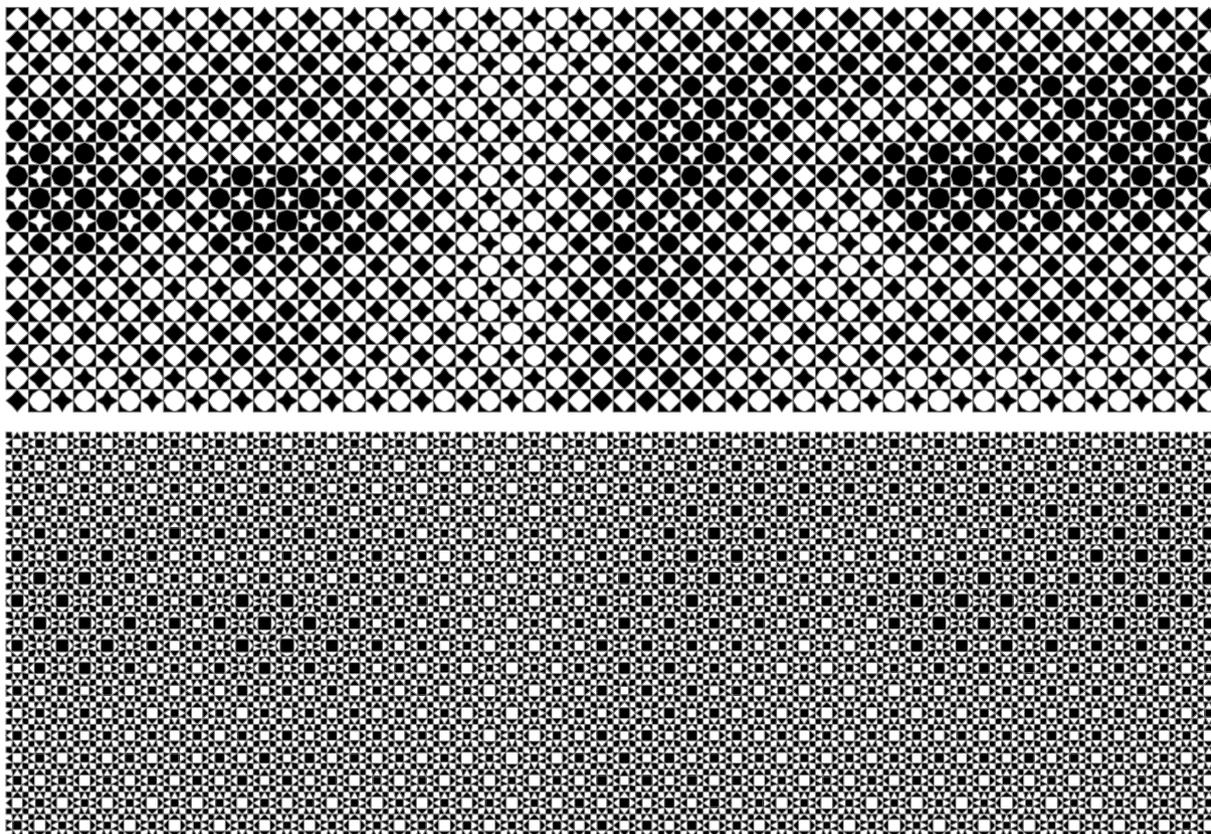


Figure 9: More two-color map-colored mosaics.

References

- [1] R. Bosch, "Opt Art," *Math Horizons*, February 2006, 6-9.
- [2] R. Bosch. Edge-constrained tile mosaics. In *Bridges Donostia: mathematical connections in art, music, and science*, pages 351-360, 2007.
- [3] R. Bosch. Connecting the dots: the ins and outs of TSP Art. In *Bridges Leeuwarden: mathematical connections in art, music, and science*, pages 235-242, 2008.
- [4] R. Bosch. Simple-closed-curve sculptures of knots and links. *Journal of Mathematics and the Arts*. 4(2):57-71, 2010.
- [5] R. Bosch and A. Herman. Continuous line drawings via the traveling salesman problem. *Operations Research Letters*. 32:302-302, 2004.
- [6] R. Bosch and A. Pike. Map-colored mosaics. In *Bridges Banff: mathematical connections in art, music, and science*, pages 139-146, 2009.
- [7] C. Browne. Duotone Truchet-like tilings. *Journal of Mathematics and the Arts*. 2(4):189-196, 2008.
- [8] R.W. Floyd and L. Steinberg. An adaptive algorithm for spacial grey scale. *Proceedings of the Society of Information Display*. 17:75-77, 1976.
- [9] C.S. Kaplan and R. Bosch. TSP Art. In *Bridges Banff: mathematical connections in art, music, and science*, pages 301-308, 2005.
- [10] C.A. Pickover. Picturing randomness with Truchet tiles. *Journal of Recreational Mathematics*. 21:256-259, 1989.
- [11] L. Wolsey, *Integer Programming*, Wiley-Interscience," 1998.

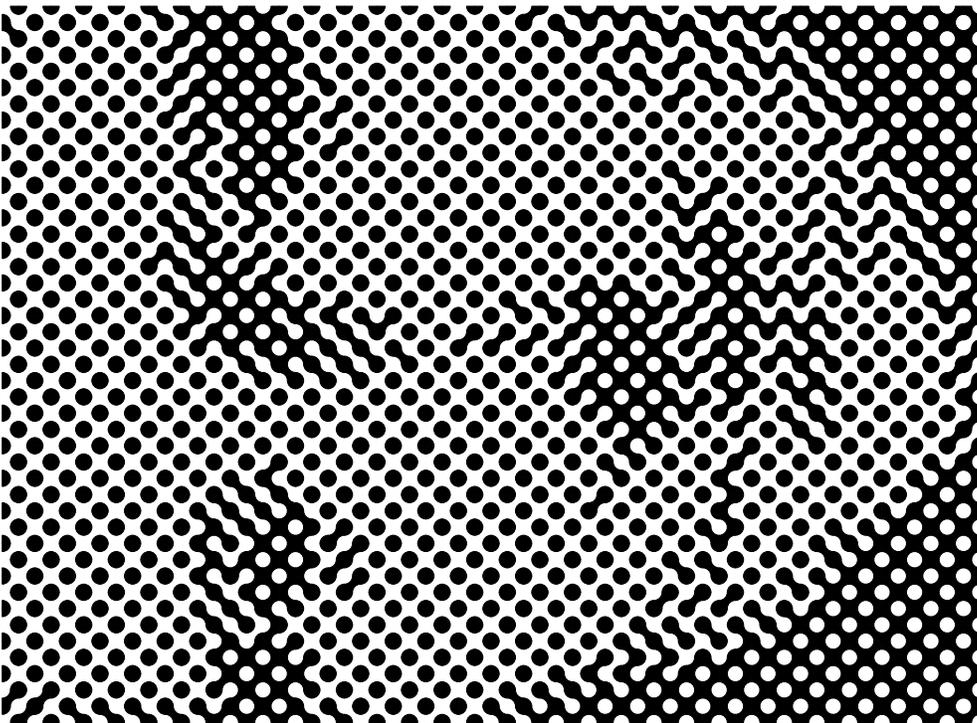
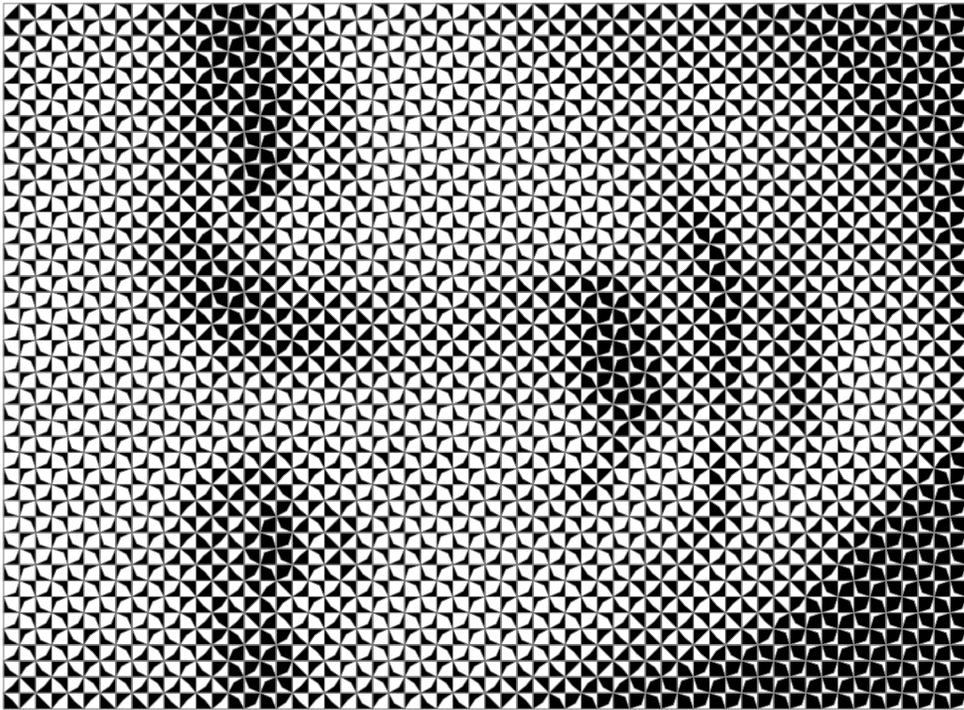


Figure 10: A portion of Leonardo da Vinci's *Mona Lisa* rendered as a two-color map-colored mosaic (top) and as an edge-matched mosaic made with black-and-white, duotone, Truchet-like tiles (bottom).