Controlling Textures in TSP Art

Robert Bosch and Robert Klock

Oberlin College, Oberlin, Ohio, USA; rbosch@oberlin.edu

Abstract

By making simple modifications to how we calculate distances between points/cities, we control textures in TSP Art.

Introduction

TSP Art [1,2,5] is produced by (1) applying a stippling algorithm to a grayscale image, (2) interpreting the resulting point set as the cities of a Traveling Salesperson Problem (TSP) [4], (3) finding a high-quality tour of the points/cities, and finally, drawing the tour. Figure 1 displays an example. The point set consists of 8192 points that collectively resemble Apple's *Water Wave Emoji*, which was based on Hokusai's *The Great Wave off Kanagawa*. The tour is a single cycle through these points, what an artist would describe as a continuous line drawing. The salesperson could start at (x_1, y_1), the point in the top left corner, then move to a second point (x_2, y_2), and then a third point (x_3, y_3), and so on. After visiting the last point, the salesperson must return directly to their starting point. The length of the ith tour segment is given by $l_i = [\sqrt{(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}]$, the Euclidean distance between the points (x_i, y_i) and (x_{i+1}, y_{i+1}), rounded to an integer, and the total length L of the tour is the sum of the l_i's. This particular tour has $L = 2097770$. We obtained the tour with the Concorde TSP Solver's [3] implementation of the Lin-Kernighan heuristic, and we also used Concorde to solve a sequence of linear programming (LP) problems that provided us with lower bounds on the length L^* of an optimal (minimum length) tour. Here, the final LP lower bound of 2097312.64 tells us that this tour is within 0.0218% of optimal.
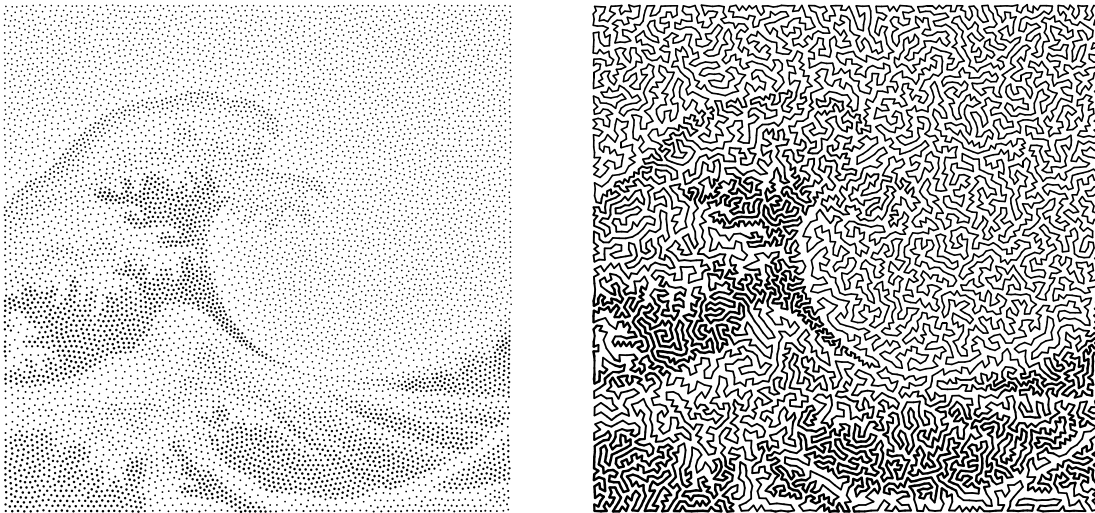
Figure 1: *(a) 8192 points that resemble Hokusai's wave, and (b) a high-quality TSP tour through the points.*

Figure 2 displays a second example. The point set contains 8192 points that collectively look like Apple's *Sun Emoji*, and the tour connects these points into a single cycle of length $L = 2081805$. The final LP lower bound is 2081293.20, which allows us to conclude that this tour is within 0.0246% of optimal.
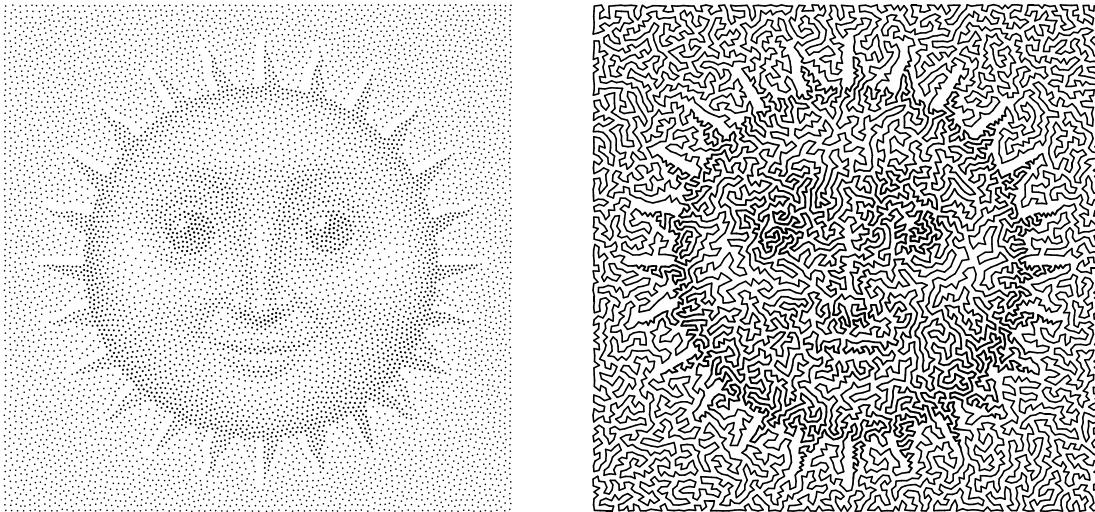

Figure 2: *(a) 8192 points that resemble a sun emoji, and (b) a high-quality TSP tour through the points.*

Two comments are in order: First, in an effort to heighten the contrast of all four images, we allowed ourselves to vary the point radii and line-segment thicknesses in accordance with the underlying grayscale target images. Second (and much more importantly), we used MacQueen's algorithm [6] to perform the stippling. The patterns formed by the "background" sections of the tours (the sky portion of the Hosukai piece, and everything but the sun in the sun piece) are characteristic of tours of point sets created with MacQueen's algorithm, patterns that may bring to mind the grooves, crinkles, and folds of brain coral.

Controlling Textures

In this paper, we describe simple methods for controlling the textures that appear in TSP Art. We do this not by altering the stippling algorithm, but by changing how we measure distances and compute the lengths of tours. Figure 3 displays (a) a point set consisting of 128 points and (b,c,d) three tours through these points.
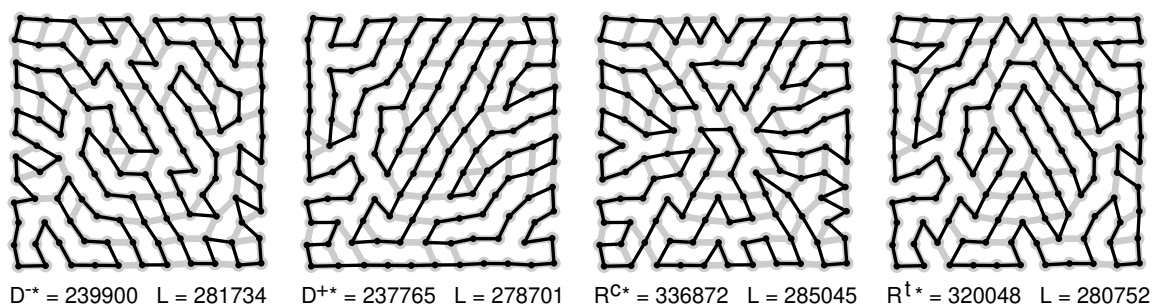

$L^* = 274839$ $\quad$ $V^* = 236315$ $\quad L = 279701$ $\quad$ $H^* = 233666$ $\quad L = 282522$
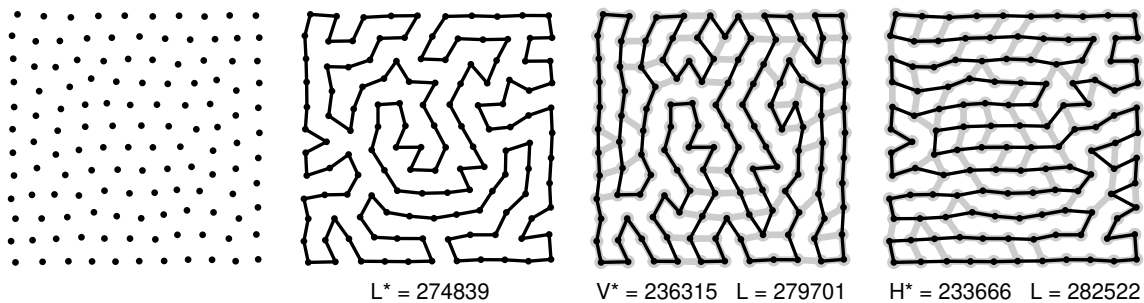
Figure 3: *(a) 128 randomly positioned points together with (b) an L-minimizing tour through them, (c) a V-minimizing tour through them, and (d) an H-minimizing tour through them.*

To obtain the tour shown in Figure 3(b), we minimized the total length L, just as we did to obtain the tours shown in Figures 1(b) and 2(b). To obtain the tour shown in Figure 3(c), we minimized an alternate length measure, V, that encourages the salesperson to use tour segments that are vertical (or somewhat close to vertical). To compute V, we used $v_i = [\sqrt{(x_{i+1} - x_i)^2 + 0.625(y_{i+1} - y_i)^2}]$ instead of l_i, to measure the length of the tour segment that joins points (x_i, y_i) and (x_{i+1}, y_{i+1}), and we set V equal to the sum of all of the v_i's. This measure is appropriate in a world in which it is easier for the salesperson to travel "vertically" (in the north and south directions) than "horizontally" (in the east and west directions). We think of 0.625 as the value of an "easing parameter." We tried various values before settling on 0.625. The smaller the value, the more incentive the salesperson has to use vertical (or somewhat vertical) tour segments.

To obtain the tour shown in Figure 3(d), we minimized a second alternate length measure, H, that incentivizes tour segments that are horizontal (or somewhat close to horizontal). To compute H, we used $h_i = [\sqrt{0.625(x_{i+1} - x_i)^2 + (y_{i+1} - y_i)^2}]$ instead of l_i, and we set H equal to the sum of all of the h_i's. This measure is appropriate if it is easier for the salesperson to travel horizontally than vertically.

We used the Concorde TSP Solver to obtain the tours. The L-minimizing tour shown in Figure 3(b) has $L^* = 274839$. The V-minimizing tour shown in Figure 3(c) has $V^* = 236315$ and $L = 279701$, which makes it 1.77% longer than the L-minimizing tour. The H-minimizing tour shown in Figure 3(d) has $H^* = 233666$ and $L = 282522$, which makes it 2.80% longer than the L-minimizing tour. Concorde required just a fraction of a second to solve the L-minimizing and H-minimizing instances, and it needed about a second and a half to solve the V-minimizing instance. Figures 3(c) and 3(d) show both the L-minimizing tour in light gray and the V-minimizing and H-minimizing tours in black.

Figure 4 displays four additional differently textured tours through these points. The tour shown in Figure 4(a) minimizes a measure D^- that incentivizes travel in the northwest and southeast directions (along the negatively sloping diagonal through the point set). To compute D^-, we started by rotating each point (x_i, y_i) in the point set $\pi/4$ radians counterclockwise about the center of the point set. Letting (x_i', y_i') stand for the rotated version of (x_i, y_i), we used $d_i^- = [\sqrt{(x_{i+1}' - x_i')^2 + 0.625(y_{i+1}' - y_i')^2}]$ instead of l_i to measure the length of the tour segment connecting points (x_i, y_i) and (x_{i+1}, y_{i+1}). Finally, we set D^- equal to the sum of all of the d_i^-'s. For the tour shown in Figure 4(b), we incentivized travel in the northeast and southwest directions (along the positively sloping diagonal) by using $d_i^+ = [\sqrt{0.625(x_{i+1}' - x_i')^2 + (y_{i+1}' - y_i')^2}]$ instead of l_i and defining D^+ to be the sum of all of the d_i^+'s.

| $D^{-*} = 239900$ $L = 281734$ | $D^{+*} = 237765$ $L = 278701$ | $R^{c*} = 336872$ $L = 285045$ | $R^{t*} = 320048$ $L = 280752$ |

Figure 4: *Four more tours: (a) NW/SE-incentivized tour, (b) a NE/SW-incentivized tour, (c) a radially outward (from center) incentivized tour, (d) a radially outward (from top center) incentivized tour.*

The tour shown in Figure 4(c) minimizes a measure R^c that encourages the salesperson to use tour segments that radiate outward from the center of the point set. To compute R^c, we started by finding (x_c, y_c), the center of the point set, and $\rho = \max_i\{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2}\}$. Then, to measure the length of the tour segment connecting points (x_i, y_i) and (x_{i+1}, y_{i+1}), we computed l_i, the Euclidean length of the tour segment, rounded to an integer, and the angle θ between the vectors $(x_i - x_c, y_i - y_c)$ and $(x_{i+1} - x_c, y_{i+1} - y_c)$. It turns

out that

$$\theta = \arccos\left(\frac{(x_i - x_c)(x_{i+1} - x_c) + (y_i - y_c)(y_{i+1} - y_c)}{\sqrt{(x_i - x_c)^2 + (y_i - y_c)^2} \sqrt{(x_{i+1} - x_c)^2 + (y_{i+1} - y_c)^2}}\right).$$

Finally, we set

$$r_i^c = [l_i + \rho\theta/\pi]$$

for each i and set R^c equal to the sum of all of the r_i^c's, rounded to integers. Note that when the angle θ is close to 0, the modified length r_i^c will be close to the non-modified length l_i. As θ increases from 0, the modified length r_i^c will increase as well, deviating more and more from the non-modified length l_i. As θ gets closer and closer to its maximum value of π, the modified length r_i^c will get closer and closer to $l_i + \rho$, which is considerably larger than the non-modified length l_i. For the tour shown in Figure 4(d), we used a similarly defined measure R^t that encourages the salesperson to use tour segments that radiate outward from (x_t, y_t), the midpoint of the top edge of the convex hull of the point set.

As before, we used Concorde to obtain the tours. And as we did in Figure 3, we show both the L-minimizing tour in light gray and the other tours in black. The D^--minimizing tour shown in Figure 4(a) has $D^{-*} = 239900$ and $L = 281734$, which makes it 2.50% longer than the L-minimizing tour. The D^+-minimizing tour shown in Figure 4(b) has $D^{+*} = 237765$ and $L = 278701$, which makes it 1.40% longer than the L-minimizing tour. The R^c-minimizing tour shown in Figure 4(c) has $R^{c*} = 336872$ and $L = 285045$, making it 3.71% longer than the L-minimizing tour. And finally, the R^t-minimizing tour shown in Figure 4(d) has $R^{t*} = 320048$ and $L = 280752$, making it 2.15% longer than the L-minimizing tour. Concorde required 2.0, 1.3, 3.4, and 1.2 seconds, respectively, to solve these instances.

Artistic Possibilities

By controlling textures, we can produce higher quality TSP Art. The texture-controlled version of the wave piece shown in Figure 5(a) uses the same point set as Figure 1. For this new version, we used the standard texture for the foreground and the H-minimizing (horizontal) texture for the background, a cloud-filled sky.
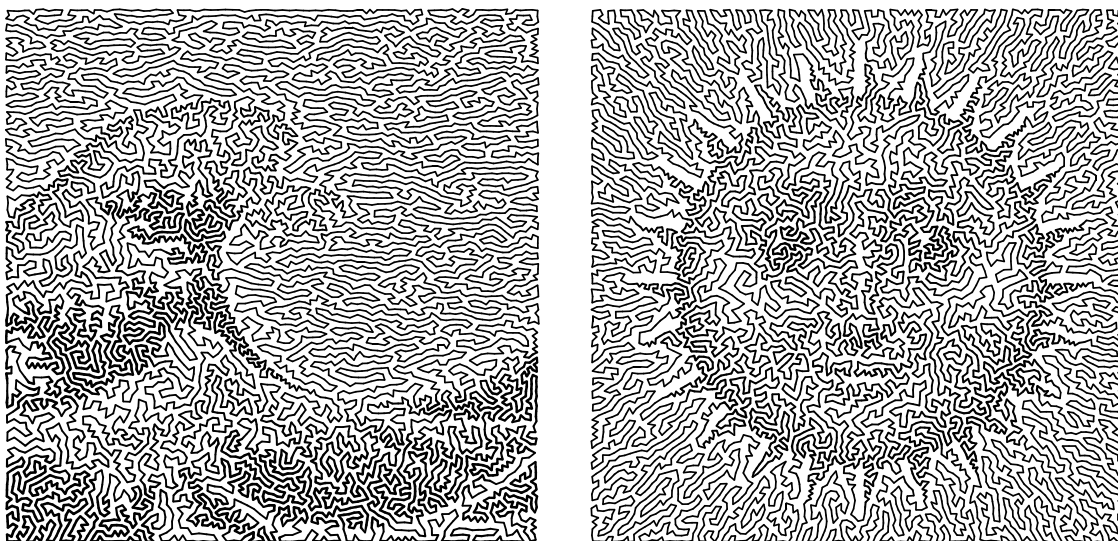
Figure 5: *Two examples of texture-controlled TSP Art.*

Similarly, the texture-controlled version of the sun piece shown in Figure 5(b) uses the same point set as Figure 2. For this new version, we used the standard texture for the foreground and the R^c-minimizing texture for the background. This background texture suggests rays of light that emanate from the sun.

Figure 6 provides two more examples and allows for easy comparison of standard and textured-controlled TSP Art. Figure 6(a) shows a standard TSP Art piece based on a van Gogh self-portrait [8], and Figure 6(b) shows a textured-controlled version that uses the standard texture for the foreground and the V-minimizing (vertical) texture for the background. Both tours pass through the same 8192-city point set. Both pieces resemble the target image, but the texture-controlled version is superior in two ways: First, the difference in textures between foreground and background helps to heighten the contrast between foreground and background. And second, the vertical background texture in the texture-controlled tour is more consistent with the wavy vertical background texture in the actual van Gogh self portrait (but with none of its vortices).
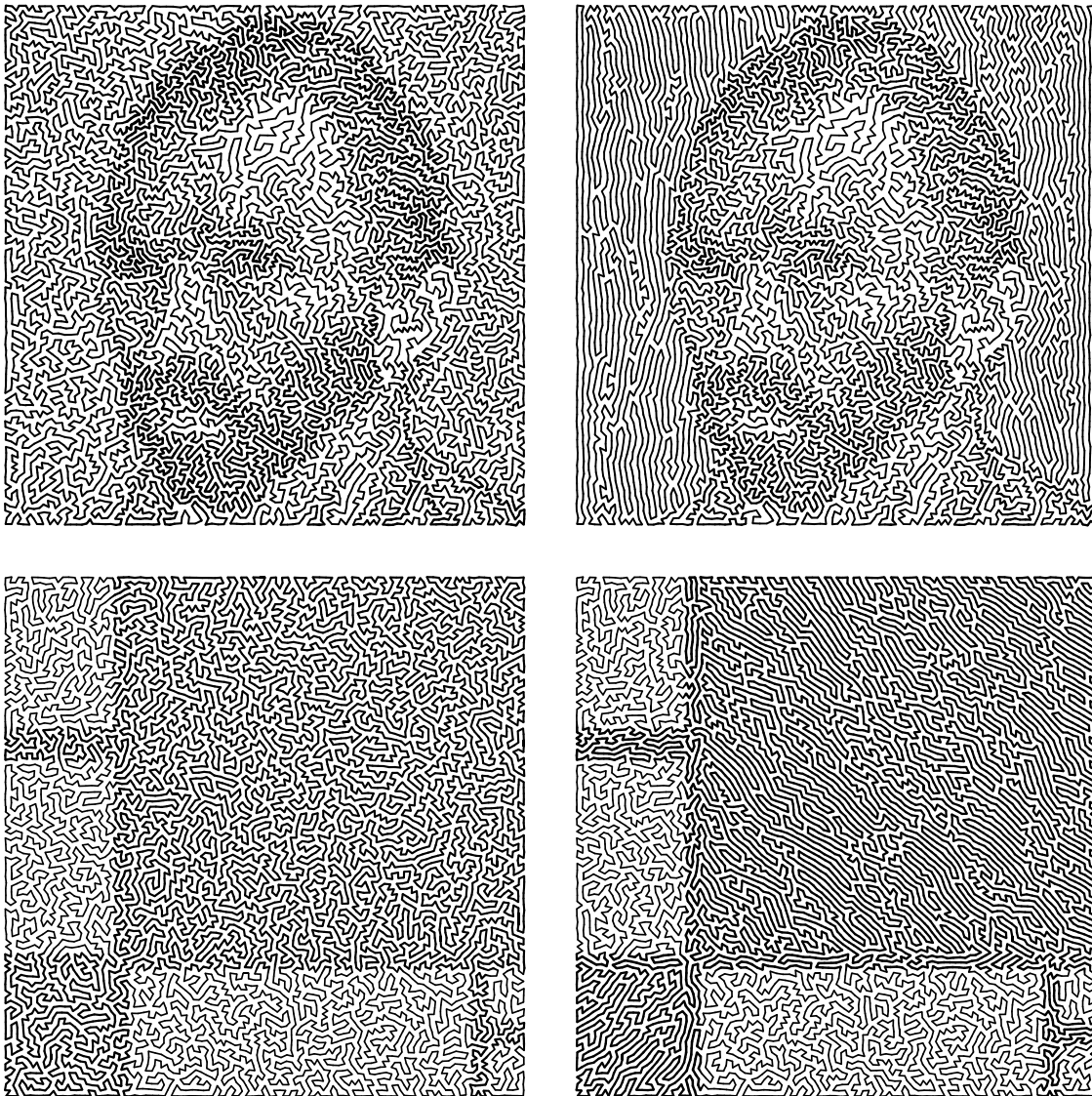
Figure 6: *Standard (a,c) versus texture-controlled (b,d) TSP Art.*

Figure 6(c) shows a standard TSP Art piece inspired by a Mondrian painting [7], and Figure 6(d) shows a textured-controlled version with five textures: the standard texture (in the brightest rectangular regions), the D^+-minimizing texture (in the bottom left corner), the D^--minimizing texture (in the top right square), the V-minimizing texture (for the vertical lines), and the H-minimizing texture (for the horizontal lines). As in the van Gogh example, both tours pass through the same 8192-city point set.

Conclusions

We have shown that by making simple modifications to how we calculate distances between points/cities, we control textures in TSP Art. All of our modifications are location based. If two points both belong to one particular region of the target image (e.g., the foreground), we use one formula to compute the distance between them. If they both belong to another (e.g., the background), we use a different formula. (If they belong to different regions, we can use the standard Euclidean distance formula.)

In the future, we plan on investigating more complicated location-based modifications. We also hope to devise modifications that take into account the brightness levels of the underlying target image. If b_i and b_{i+1} denote the brightness levels that correspond to points i and i+1 (on a 0-to-1, black-to-white grayscale) and $0 < |\alpha| < 1$, then we could use $l'_i = (1 + \alpha|b_i - b_{i+1}|)l_i$ in place of l_i. If $\alpha < 0$, then when the salesperson is at point i, they will have a preference—all else being equal—for moving to a point that has a substantially different brightness level than point i. If $\alpha > 0$, they will prefer to move to a point that has the same (or nearly the same) brightness level as point i.

Acknowledgements

References

[1] Robert Bosch. *Opt Art: From Mathematical Optimization to Visual Design*. Princeton University Press, 2019.

[2] Robert Bosch and Adrianne Herman. Continuous line drawings via the traveling salesman problem. *Operations Research Letters*, vol. 32. no. 4, pages 302-303, 2004.

[3] Concorde TSP Solver. http://www.math.uwaterloo.ca/tsp/concorde.html

[4] William J. Cook. *In Pursuit of the Traveling Salesman: Mathematics at the Limits of Computation*. Princeton University Press, 2012.

[5] Craig S. Kaplan and Robert Bosch. TSP art. In Reza Sarhangi and Robert Moody, editors, *Bridges Conference Proceedings*, Banff, Alberta, Canada, July 31–Aug. 3, 2005, pages 301-308. https://archive.bridgesmathart.org/2005/bridges2005-301.html

[6] J. MacQueen. Some methods for classification and analysis of multivariate observations. In Lucien M. Le Cam and Jerzy Neyman, editors, *Proceedings of the 5th Berkeley Symposium on Mathematical Statistics and Probability, Vol. 1*, pages 281-297, Berkeley, California, 1967. University of California Press.

[7] Piet Mondrian. *Composition with Red, Blue and Yellow*, 1930. Oil on canvas. Musée d'Orsay, Paris.

[8] Vincent van Gogh. *Self-Portrait*, 1889. Oil on canvas. Kunsthaus Zürich, Zurich.