

Inferring compositional style in the neo plastic paintings of Piet Mondrian by machine learning

David Andrzejewski,^a David G. Stork,^b Xiaojin Zhu^a and Ron Spronk^c

^aDepartment of Computer Sciences, University of Wisconsin-Madison, Madison WI 53706

^bRicoh Innovations, 2882 Sand Hill Road Suite 115, Menlo Park CA 94025 USA

^cDepartment of Art History, Queen s University, Kingston, ON K7L, Ontario Canada

ABSTRACT

We trained generative models and decision tree classifiers with positive and negative examples of the neo-plastic works of Piet Mondrian to infer his compositional principles, to generate “faux” works, and to explore the possibility of computer-based aids in authentication and attribution studies. Unlike previous computer work on this and other artists, we used “earlier state” works—intermediate versions of works created by Mondrian revealed through x-radiography and infra-red reflectography—when training our classifiers. Such intermediate state works provide a great deal of information to a classifier as they differ only slightly from the final works. We used methods from machine learning such as leave-one-out cross validation. Our decision tree classifier had accuracy of roughly 70% in recognizing the genuine works of Mondrian versus computer-generated replicas with similar statistical properties. Our trained classifier reveals implicit compositional principles underlying Mondrian’s works, for instance the relative visual “weights” of the four colors (red, yellow, blue and black) he used in his rectangles. We used our trained generative model to generate “faux” Mondrians, which informally possess some of the compositional attributes of genuine works by this artist.

Keywords: Piet Mondrian, stylometry, neo-plastic painting, abstract art, composition principles, machine learning

1. INTRODUCTION

The neo-plastic paintings of Piet Mondrian (1872–1944) are spare yet subtle abstract compositions of horizontal and vertical black lines with abutting rectangles of uniform red, yellow, blue and black on fields of white or off-white. These works eschew traditional content of subject, cultural references and the natural world and instead exploit a vocabulary of pure abstraction of the artificial; there are no “natural” colors of green or brown, for example, no curves, no modulation in lightness, no explicit rendering of depth. He and other members of De Stijl (Dutch: “the style”), such as Theo van Doesburg, Vilmos Huszàr and Bart van der Leck, focused on what they felt were art’s essential elements, and explored these compositional elements over decades.

The spare nature of Mondrian’s works has attracted much compositional analysis by art historians.^{1,2} Moreover, for nearly a half century, computer scientists have also analyzed these paintings, as well works by other abstract painters, such as the Ocean Park paintings by the American Abstractionist Richard Diebenkorn,^{3,4} though with only modest empirical success and little, if any, affect upon the broader community of art scholars. Most of these computational efforts were empirical or *ad hoc* searches for compositional rules rather than principled inference using the techniques from statistical learning theory and none (to our knowledge) used examples of Mondrian’s earlier state compositions, as recently revealed through x-radiography and infrared reflectography.⁵

There are at least three reasons, then, to revisit the problem of computer methods for the analysis and synthesis of Mondrian’s works:

Send correspondence to David G. Stork, artanalyst@gmail.com.

Progress in machine learning: Early computer research focussed on hand-crafted rules for creating works in the style of Mondrian and the performance of the algorithms was judged based on expert judgements of the final works.³ Hand-crafted rules often express the bias of their creator, and lead to rules that are ultimately based on subjective opinion rather than objective, reproducible evidence. In the past several decades, though, there has been a great deal of research in machine learning algorithms and empirical success in real-world problems.⁶ Below we apply such machine learning methods with pruning, regularization, and so forth. Moreover, these methods allow us to make principled, empirical comparisons of classification methods (for example through classification rate on unknown works), rather than informal and possibly conflicting human aesthetic and artistic judgements.

New evidence of Mondrian’s working methods: Central to algorithms of machine learning are training data of positive and negative examples and especially valuable are “near misses”—here, paintings that are quite similar to genuine Mondrian paintings, but differ somewhat from his final works. Recently, x-ray and infra-red reflectography reveal earlier state versions of several of Mondrian’s works—versions that in some cases he considered developed but nevertheless painted over.⁵ Such compositions are extremely informative because we can consider them as lying on the compositional boundaries of his oeuvre.

Success of computer methods and growing acceptance by art scholars: There is growing scholarship in the interdisciplinary field of computer vision and image analysis of art as adjuncts to traditional art historical research.^{7,8} For instance, there have been a number of modest successes in the nascent subfield of *stylometry*—the mathematical description of an artist’s style (be it in brush stroke, color, composition, etc.), successes that have garnered the interest of traditional art scholars. For instance, image processing and pattern recognition algorithms applied to images of brush strokes and dripped paint can be used for attribution studies and to reveal the number of different artists or “hands” in a given work.^{9–11} The successes, while modest at present, range from authenticating works such as those by Vincent van Gogh, the analysis of brush strokes¹¹ and drips,¹⁰ and even rigorous tests for artists’ use of optical aids.¹² Computer methods may provide quantification of aspects of style (color palette, brush stroke style, etc.) for diachronic studies of an artist’s oeuvre.

We developed feature extraction software for extracting horizontal and vertical line segments, including their widths, and the parameters of the colored rectangles from images of Mondrian’s paintings, and a normalized list representation. We explored two generative and classification models: The first model was standard decision tree, in which the simple rules in each node were likewise learned from 45 positive and a variable number of negative examples. The rules learned in our decision tree classifier could be interpreted, for instance as the relative visual “weights” of colors for visual balance. Such extracted rules may be of interest to art scholars. The second model was a generative model with parameters estimated from true Mondrians using maximum likelihood estimation (MLE).⁶ We used this model to *generate* “faux” Mondrians, some of which bore some similarity to genuine Mondrians, though they lacked subtleties of genuine works, such as subtle visual “rotation.” Our work extends ongoing research in stylometry (the mathematical description of artistic style) from low-level properties such as color and brush stroke, to high-level compositional principles.

Section 2 describes Mondrian’s neo-plastic paintings such as we analyze and some of the context of their creation. Section 3 describes our automatic image feature extraction methods the internal representation we use for learning. Sections 4 and 5 describe our machine learning methods and classification results. We then display some “faux” Mondrian images generated by our statistical model. We conclude in Sect. 6 with a discussion of some of the lessons learned and future directions in the computer analysis of style in art.

2. IMAGE DATA

We restricted our analyses to Mondrian’s classical neo-plastic compositions in oil on canvas. These abstract works eschew reference to cultural or natural sources and contain horizontal and vertical black lines and rectangles of black, yellow, red and blue on fields of white and off-white. We did not include his few works that contained diagonal lines or were on diamond canvases, such as his *Tableau No. IV: Lozenge Composition with Red, Gray, Blue, Yellow, and Black* (c. 1924–25), 142.8 × 142.3 cm, or *Victory Boogie Woogie* (1942–44) 127 × 127 cm,

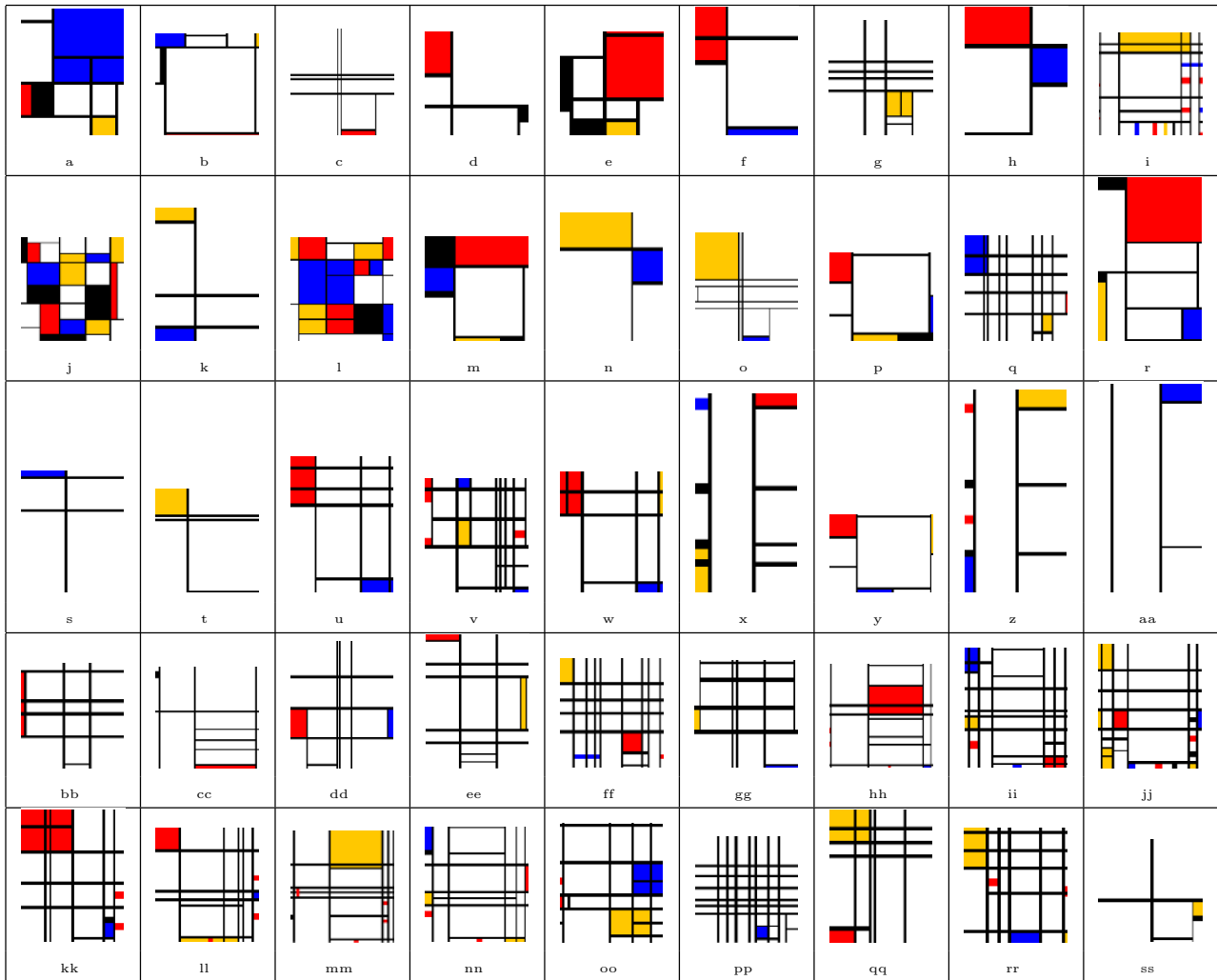


Figure 1. a) *Composition with Large Blue Plane, Red, Black, Yellow, and Gray* (1921), 60.5 × 50 cm, b) *Composition with Blue, Yellow, Black, and Red* (1922), 53 × 54 cm, c) *Composition No I, Gris-Rouge* (1935), 56.9 × 55 cm, d) *Composition No I, with Red and Black* (1929), 52 × 52 cm, e) *Tableau with Large Red Plane, Blue, Black, Light Green, and Grayish Blue* (1921), 49.5 × 49.5 cm, f) *Composition with Red and Blue* (1933), 41.2 × 33.3 cm, g) *Composition with Yellow* (1936), 74 × 66 cm, h) *Composition No II, with Red and Blue* (1929), 40.3 × 32.1 cm, i) *Place de la Concorde* (1938–43), 94 × 94.4 cm, j) *Composition A* (1920), 90 × 91 cm, k) *Composition B) en Bleu, Jaune et Blanc* (1936), 43.5 × 33.5 cm, l) *Composition C* (1920), 60.3 × 61 cm, m) *Composition No III/Fox-Trot B, with Black, Red, Blue, and Yellow* (1929), 45.4 × 45.4 cm, n) *Composition with Yellow and Blue* (1932), 41.3 × 33 cm, o) *Composition No II) Bleu-Jaune* (1935), 72.3 × 69.2 cm, p) *Composition of Red, Blue, Yellow and White: Nom III* (1939), 44.6 × 38.2 cm, q) *Composition with Blue, Red, and Yellow* (1935–42), 72.2 × 69.5 cm, r) *Tableau I, with Black, Red, Yellow, Blue, and Light Blue* (1921), 96.5 × 60.5 cm, s) *Composition with Double Line and Blue* (1934), 60 × 50 cm, t) *Composition with Double Line and Yellow* (1932), 45.2 × 45.2 cm, u) *Composition with Red and Blue* (1939–41), 43.5 × 33 cm, v) *Composition with Red, Blue, and Yellow* (1937–42), 60.3 × 55.2 cm, w) *Composition with Blue, Red, and Yellow* (1935–42), 100 × 50.5 cm, x) *Composition with Red, Blue, Yellow, Black, and Gray* (1922), 41.9 × 48.6 cm, y) *Composition with Red, Yellow, and Blue* (1927), 40 × 52 cm, z) *Composition with Red, Yellow, and Blue* (1935–1942), 101 × 51 cm, aa) *Composition Blanc et Bleu* (1936), 121.3 × 59 cm, bb) *Composition-Blanc et Rouge: B* (1936), 51.5 × 50.5 cm, cc) *Composition en Blanc, Noir et Rouge* (1936), 102 × 104 cm, dd) *Composition en Blanc, Rouge et Bleu* (1936), 98.5 × 80.3 cm, ee) *Composition-Blanc, Rouge et Jaune: A* (1936), 80 × 62.2 cm, ff) *Composition with Yellow, Blue, and Red* (1939–42), 72.5 × 69 cm, gg) *Composition en Jaune, Bleu et Blanc: I* (1937), 57.1 × 55.2 cm, hh) *Composition No 4, with Red and Blue* (1938–42), 100.3 × 99.1 cm, ii) *Composition No 5, with Blue, Yellow, and Red* (1939–42), 75 × 65 cm, jj) *Trafalgar Square* (1939–43), 145.2 × 120 cm, kk) *Composition No 7, with Red and Blue* (1937–42), 80.5 × 62.2 cm, ll) *Composition No 8, with Red, Blue, and Yellow* (1939–42), 75.2 × 68.1 cm, mm) *Composition No 9, with Yellow and Red* (1939–42), 79.7 × 74 cm, nn) *Composition No 10, with Blue, Yellow and Red* (1939–42), 79.5 × 73 cm, oo) *Composition No 11, with Blue, Red, and Yellow* (1940–42), 82.5 × 71.1 cm, pp) *Composition No 12, with Blue* (1936–42), 62 × 60.5 cm, qq) *No I: Opposition de Lignes, de Rouge et Jaune* (1937), 43.5 × 33.5 cm, rr) *Picture II, with Yellow, Red, and Blue* (1936–43), 60 × 55 cm, ss) *Composition with Yellow* (1930), 46 × 46.5 cm.

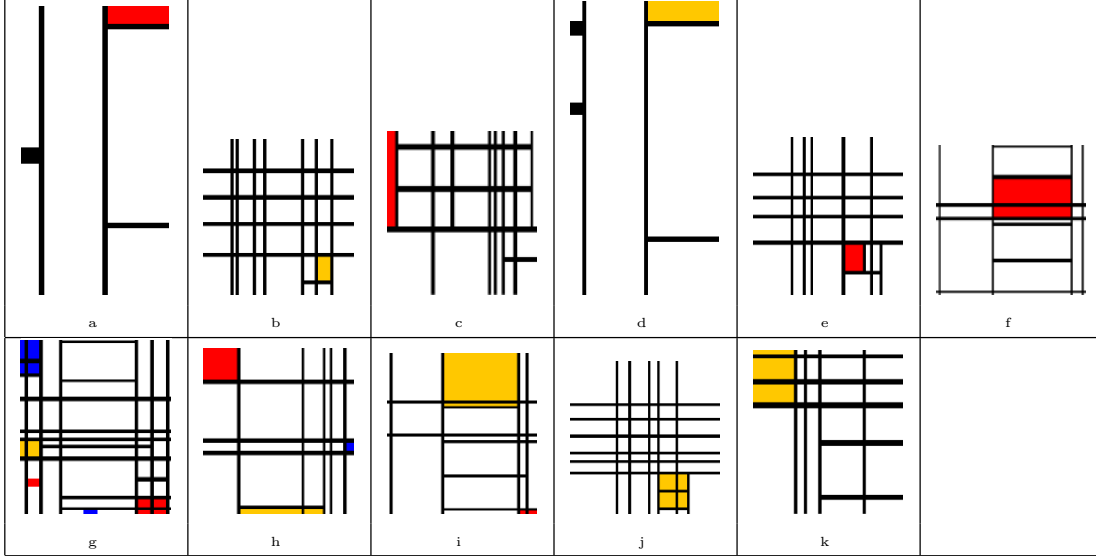


Figure 2. A set of “earlier state” works, which we refer to here as “es,” by Mondrian, that is, versions of works later altered or painted over. These images were revealed through infrared reflectography or x-ray imaging. a) *Composition with Blue, Red, and Yellow es* (1935–42), 100 × 50.5 cm, b) *Composition with Blue, Red, and Yellow es* (1935–42), 72.2 × 69.5 cm, c) *Composition with Red, Blue, and Yellow es* (1937–42), 60.3 × 55.2 cm, d) *Composition with Red, Yellow, and Blue es* (1935–42), 101 × 51 cm, e) *Composition with Yellow, Blue, and Red es* (1939–42), 72.5 × 69 cm, f) *Composition No 4, with Red and Blue es* (1938–42), 100.3 × 99.1 cm, g) *Composition No5, with Blue, Yellow, and Red es* (1939–42), 75 × 65 cm, h) *Composition No8, with Red, Blue, and Yellow es* (1939–42), 75.2 × 68.1 cm, i) *Composition No9, with Yellow and Red es* (1939–42), 79.7 × 74 cm, j) *Composition No12, with Blue es* (1936–42), 62 × 60.5 cm, k) *Picture II, with Yellow, Red, and Blue es* (1936–43), 60 × 55 cm.

as these present a number of difficulties in interpretation and representation beyond the scope of our research. Figures 1 and 2 show our computer “sketches” of the Mondrian works we considered, that is, somewhat idealized digital images generated from the representations extracted from digital images of the originals, as described in Sect. 3. We are well aware that subtleties in color variation, visual weight due to brush stroke texture and so forth are not captured by our method (cf. Sect. 6).

3. FEATURE EXTRACTION AND IMAGE REPRESENTATION

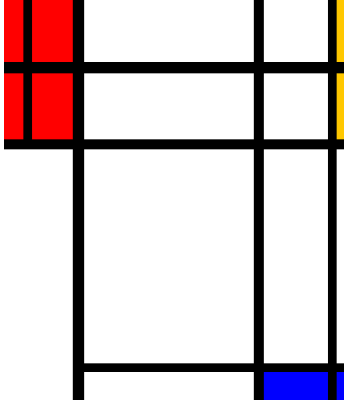
Our pattern recognition and machine learning algorithms operate on a compressed representation.⁶ We processed colored, digital images of each painting to extract lines and colored rectangles, and converted this image information into our compressed representation.

3.1 Feature extraction

Our feature extraction software uses well-known edge and line extraction algorithms modified only by the incorporation of the constraint that each line was horizontal or vertical.¹³ Likewise, the extraction of the colored rectangles incorporated the prior constraint (for color clustering) that each rectangle was either black, red, yellow or blue, and that its edges coincided with the edge of a line or the boundary of the canvas. The extraction software combines these automated methods with manual human intervention to correct extraction errors.

3.2 Image representation

Pattern recognition on a small or moderate-sized data set requires that the patterns be represented in a compressed format—ones that can represent all the training patterns yet not be so general as to easily represent patterns unlike those in the training set. The candidate representations need not be unique, but the smaller the



```

ymax 600
xmax 512
vpts [1 36 112 382 492 512]
hpts [1 102 217 552 600]
rectcolors [2 2 1 1 3 2 2 1 1 3 1 1 1 1 1 4 4]
vext [1 5, 1 3, 1 5, 1 5, 1 5, 1 5]
vthick [0 11 15 14 12 0]
hext [1 6, 1 6, 1 6, 3 6, 1 6]
hthick [0 16 14 11 0]
rect [1 2 1 2, 2 3 1 2, 3 4 1 2, 4 5 1 2, 5 6 1 2,
1 2 2 3, 2 3 2 3, 3 4 2 3, 4 5 2 3, 5 6 2 3,
1 3 3 5, 3 4 3 4, 4 5 3 4, 5 6 3 4, 3 4 4 5,
4 5 4 5, 5 6 4 5]

```

Figure 3. Computer sketch of Mondrian’s *Composition With Red Blue Yellow And White: Nom III* (1939) 44.6 38.2 cm, and key aspects of its representation in our framework.

training set the more important is the creation of a good internal representation. In some classification problems, a good representation is one that matches the generative representation, that is, the fundamental units that underly the creation of the patterns. Clearly representations based on pixel-level features are inappropriate.

The fundamental building blocks for our image representation were *grid points*. Each grid point is duple, defined by a single numerical coordinate, and an attribute specifying whether this coordinate is horizontal or vertical. Thus $(V, 31)$ defines a vertical grid point located at pixel row 31. The locations of all image components—lines, rectangles—can then be described in terms of these grid points.

- The borders of the image are defined as four initial grid points
 - $\{(V, 0), (V, y_m), (H, 0), (H, x_m)\}$
- Horizontal lines are described with three grid points
 - The height of the center of the line is defined with a vertical grid point (V, y_{center})
 - The extent of the line is defined with two horizontal grid points $\{(H, x_{start}), (H, x_{end})\}$
- Vertical lines are defined analogously, swapping horizontal and vertical grid points
- Colored rectangles are defined by two horizontal and two vertical grid points
 - $\{(V, y_{top}), (V, y_{bottom})\}$
 - $\{(H, x_{left}), (H, x_{right})\}$

Colored rectangles have an additional color attribute $c \in \{R, G, B, K\}$, while lines will have an additional thickness (in pixels) attribute $t \in \mathcal{Z}$. Lines are assumed to be black and all space not filled by colored rectangles or lines is assumed to be background (white). In addition to the four initial grid points, an image will only require the definition of grid points corresponding to the centers of all horizontal and vertical lines. If necessary, additional grid points can be defined for the special case of rectangles which have non-line borders such as the edge of the canvas.

An advantage of the grid point representation is that it can directly capture the fact that horizontal lines often start and end at vertical lines, and vice versa. For example, the horizontal grid point specifying the center of a vertical line may also be the endpoint of a horizontal line. Lines that start or end on image borders can be represented in this way using the four initial “border” grid points. Also, note that in our representation the lines are naturally represented as being drawn *on top of* the colored rectangles. This allows our representation

to express adjacent rectangles as a single rectangle instance with line(s) crossing over them, a more compressed representation. Moreover, such color rectangles are often perceived by viewers as unitary rather than as abutting rectangles, as can be seen for instance in the red rectangles in figures f), u) and kk), and the yellow rectangles in mm), oo) and rr) in Fig. 1 and all the colored rectangles in Fig. 3, below.

- the coordinate convention is matrix (row, column) style, that is, the upper-left corner is (0, 0), the lower-right is (600, 512)
- vpts are grid points where vertical lines are ordered left-to-right
- “ext” is short for “extent” and defines start and end grid points. For example, vext[1,1:2] specifies the start or end and hpts for the first vertical line
- rectangles are defined in terms of grid points [vleft, vright, htop, hbottom]
- the rectangle numbering scheme runs left-to-right, top-to-bottom
- colors are represented: 1 = white, 2 = red, 3 = yellow, 4 = blue, 5 = black

A Mondrian painting and key portions of its representation are illustrated in Fig. 3.

4. MACHINE LEARNING

We used two popular machine learning techniques: decision trees and generative models. We stress that in choosing these techniques and their associated data representations, we are in no way assuming that these correspond to Mondrian’s internal representation or his forward or generative model. For instance, a decision tree corresponds to a sequence of feed-forward decisions that bear scant similarity to Mondrian’s working methods of testing, revising and altering his compositions during the execution of his works. Nevertheless, these models may yield valuable insights into the stylistic properties of Mondrian paintings.

4.1 Decision trees

Binary decision trees are one of the most popular algorithms in machine learning and pattern recognition, in part because tree training algorithms are simple, the final classifier is usually interpretable, and the trees admit a number of natural regularization or complexity adjustment techniques.^{6,14} In brief, decision tree learning finds the single decision that splits the training data into two sets that are the most “pure.” Such a set (often called a “split”) is more pure when most of the examples from a single category. Then the algorithm finds the best decision for each of these subsets, so as to make *its* subsets more pure. The process is iterated until no more split are necessary and thus all subsets are pure. Finally, in order to avoid the well-known problem of overfitting, the technique of *pruning* is applied. Pruning may use additional validation data to simplify the learned model, removing splits to improve classifier performance on the validation data.

In order to apply standard decision trees, we must further convert our compressed representations into fixed-length feature vectors. We accomplished this by computing a representative features such as number of horizontal lines, ratio of horizontal to vertical lines, and the presence or absence of “composite” rectangles formed by sets of adjoining colored rectangles. We also include spatial features of the painting, such as the visual “center of mass” and the proportion of colored rectangles that have at least one edge on the border of the painting. These features in particular are meant to capture some higher-level compositional aspects of the paintings.

4.2 Training generative models

Generative models are particularly useful for both understanding the underlying distribution of data and for generating new synthetic examples from the distribution.³ We define our generative model procedurally:

1. Sample the aspect ratio from a kernel density estimate
2. Sample the number of horizontal lines $n_h \sim Poisson(\lambda_h)$

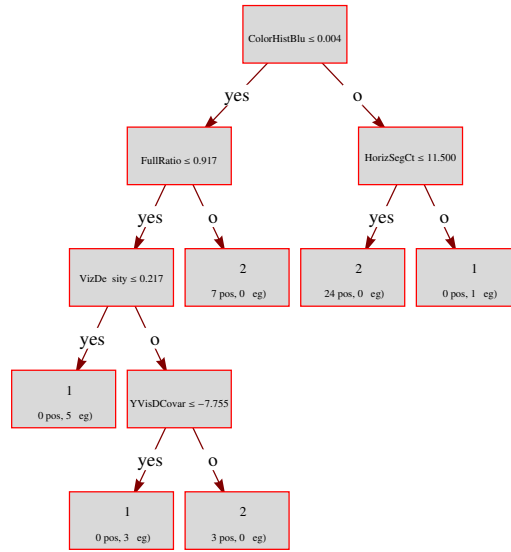


Figure 4. A binary decision tree—where in the terminal or leaf nodes 1 represents genuine Mondrians and 2 non-Mondrians—learned from examples of true and faux Mondrians.

3. Sample s from $\{-1, +1\}$, each with prob 0.5
4. Sample the difference $d \sim \text{Poisson}(\lambda_d)$
5. Let the number of vertical lines be $n_v = n_h - 2 + s * d$
6. Sample the spacing of horizontal lines $\theta_h \sim \text{Dirichlet}(\alpha_h)$
7. Sample the spacing of vertical lines $\theta_v \sim \text{Dirichlet}(\alpha_v)$
8. For all line segments, sample the state from a multinomial over $\{\textit{present}, \textit{deleted}, \textit{invisible}\}$
9. For all rectangles, sample the color from a Dirichlet-compound multinomial over $\{\textit{red}, \textit{yellow}, \textit{blue}, \textit{black}\}$

We found that many images generated by this model did not resemble Mondrians. For example, the random segment deletion model could cause remaining segments to “hang,” with one endpoint unattached to another line, rectangle or canvas edge. Likewise, generated images might have no vertical lines whatsoever. Since both of these cases never occur in the dataset, we simply enforce that only “legal” segments are deleted or made invisible during the generative procedure, and that any generated image contains at least one vertical line.

Note that all quantities in the model (the number of horizontal lines, etc.) are fully observed, assuming that the model does not allow any completely invisible or deleted lines. Combined with the fact that the most of the distributions in our model belong to the exponential family, this means that it is relatively straightforward to estimate the generative model parameters from a set of paintings. We estimated these parameters via maximum likelihood estimation⁶ using our set of 45 true Mondrians.

5. RESULTS

We now turn to our results and analyses. We begin with the classification accuracy, then analyze color information in a trained tree classifier, and finally generate “faux” using a trained generative model.

Parameter	MLE	Description
σ	0.07	Aspect ratio (kernel bandwidth)
λ_h	6.38	Number of horiz lines (Poisson)
λ_d	1.64	Difference in horizontal versus vertical lines (Poisson)
v	1.80	Vertical line spacing (Dirichlet)
h	1.61	Horizontal line spacing (Dirichlet)
p	9.61	Segment present (Dirichlet-compound multinomial)
d	4.36	Segment deleted (Dirichlet-compound multinomial)
i	0.37	Segment invisible (Dirichlet-compound multinomial)
R_w	0.754	Rect white (multinomial)
R_r	0.085	Rect red (multinomial)
R_y	0.062	Rect yellow (multinomial)
R_b	0.065	Rect blue (multinomial)
R_k	0.034	Rect black (multinomial)

Table 1. Parameters for the generative model.

5.1 Classification accuracy

There are two questions we wish to answer. First, can decision trees distinguish between true Mondrians and “faux” paintings generated by our model? Second, can decision trees distinguish between the final versions of Mondrian paintings and the “earlier states” of the TransAtlantic paintings? We evaluate these questions using leave-one-out cross-validation experiments, where the decision tree is repeatedly trained on all examples except one and then used to classify the single held-aside test example. Repeating this procedure for each individual for every example is a popular and effective technique for evaluating the performance of classifiers.⁶ Furthermore, as a baseline we compare the classification accuracy of our decision trees against a simple majority classifier which always assigns the label of the more populous class. As discussed previously, pruning requires an additional validation data set, which we obtain by setting aside 20% of the training data for every fold.

To answer the first question, we trained our generative model on the dataset of 45 true Mondrian paintings, learning the model parameters via maximum likelihood estimation. We then generated 45 synthetic paintings from our model and ran cross-fold validation experiments with decision trees on the two sets of paintings. The results are shown in Table 1. Since the 80/20 data set split in the tuning introduces some randomness, each pass over the entire dataset was repeated 10 times. The reported accuracy is the mean value over these 10 trials, and the standard deviation is also supplied. Note that a baseline approach of classifying all test examples as positive (true Mondrian) or negative (generated painting) would obtain an accuracy of 50%, while the performance of our decision trees is much better. Therefore, we can indeed distinguish between true Mondrians and generated paintings, even though the parameters of the generative model were fit using the true Mondrians.

Next, we repeat this procedure using the TransAtlantic “earlier states” as our negative class.⁵ Due to the fact that we have more true Mondrians than earlier states, we now have what is known as a *skewed* dataset. A simple majority classifier would now classify all test examples as positive (true Mondrian). On this data, decision trees are not capable of improving on the accuracy of the majority baseline. There are two possible explanations for this. One is that either our simplified representation or our classification model is simply unable to capture the differences between these two sets of paintings. Perhaps a richer representation or set of classifiers would be able to successfully discriminate between final Mondrians and earlier states. Another explanation is that there is no fundamental stylistic difference between the earlier states and the final paintings, which would be consistent with the claim that Mondrian viewed these “earlier state” paintings as finished.

5.2 Color weights

Some types of trained classifiers such as the nearest-neighbor classifiers and kernel density classifiers reveal little if any information about the fundamental feature groupings that predict category membership. This limitation is occasionally the basis of recommendations against using neural network classifiers in domains, such as medical

Pattern set	Accuracy	
	with pruning	without pruning
TransAtlantics (default color weights)	0.756 ± 0.03	0.719 ± 0.0
TransAtlantics (no “visual center” features)	0.753 ± 0.05	0.702 ± 0.0
TransAtlantics (learned color weights)	0.754 ± 0.04	0.719 ± 0.0
Synthetics (default color weights)	0.703 ± 0.68	0.681 ± 0.0
Synthetics (no “visual center” features)	0.685 ± 0.05	0.714 ± 0.0
Synthetics (learned weights)	0.677 ± 0.05	0.703 ± 0.0

Table 2. Leave-one-out decision tree classification results.

diagnosis, where the user (typically a medical doctor) needs to understand the reason for a particular computer-based diagnosis. In contrast, generative models and decision trees do typically reveal useful information.

We are particularly interested in understanding the roles of the four colors, red (r), yellow (y), blue (b) and black (k). For each rectangle we consider its mass m to be proportional to the area of the rectangle and we define the coordinates of its center as $\mathbf{c} = (c^x, c^y)$. For each color $v \in \mathcal{C} = \{r, y, b, k\}$ we average the centers and sum the masses for all rectangles of that color to obtain (c_v, m_v) . Given color weights $\{w_r, w_y, w_b, w_k\}$ the overall visual center of the image is then calculated as

$$\begin{aligned}
 c^x &= \frac{\sum_{v \in \mathcal{C}} c_v^x m_v w_v}{\sum_{v \in \mathcal{C}} m_v w_v} \\
 c^y &= \frac{\sum_{v \in \mathcal{C}} c_v^y m_v w_v}{\sum_{v \in \mathcal{C}} m_v w_v}.
 \end{aligned} \tag{1}$$

In order to learn the weights from a set of true Mondrians, \mathcal{M} , we assume that these paintings are visually “centered” at $\mathbf{c} = (0.5, 0.5)$ in normalized $(x, y) \in [0, 1]^2$ coordinates. That is, we want $(c^x, c^y) = (0.5, 0.5)$ for each of the paintings in our set of true Mondrians. Some algebraic manipulation yields the linear objective function

$$\min \left\{ \sum_{m \in \mathcal{M}} 0.5 \sum_{v \in \mathcal{C}} m_v w_v - \sum_{v \in \mathcal{C}} c_v^y m_v w_v + 0.5 \sum_{v \in \mathcal{C}} m_v w_v - \sum_{v \in \mathcal{C}} c_v^x m_v w_v \right\}. \tag{2}$$

Adding the constraints that the weights be non-negative and sum to one, we have a well-defined linear program which can be easily solved by standard optimization software packages. Using our set of 45 true Mondrians, the resulting color weights are:

$$r = 0.237; \quad y = 0.143; \quad b = 0.227; \quad k = 0.392. \tag{3}$$

Figure 5 shows squares of equal total “weight,” which is proportional to the visual weight of its color times the area of the square, as inferred from Mondrian’s neo-plastic paintings. Thus the smallest square (black) corresponds to the color with the highest weight. Informally, these weights correspond very nicely to our visual intuitions, with black being the heaviest color and yellow being the lightest.



Figure 5. Squares with equal total color weights within genuine Mondrian paintings, where here the area is inversely proportional to its color’s visual weight.

5.3 Images generated

Many types of trained models can be used to generate patterns. For instance, Grebert et al. trained an “inverted” neural network with examples of letterforms from a subset of the full alphabet and then used the network to create other letterforms—ones that exploited the learned representations for both the individual letter and the style.¹⁵ Analogously, our trained generative model can be used to generate “faux” examples. To generate images, we simply follow the generative procedure outlined earlier, using our estimated parameter values for each of the sampling steps. Some example generated images are shown in Figure 6. Clearly these generated works would not be mistaken for genuine Mondrian paintings (cf., Sect. 6)).

The clear fact that the images created by our simple generative model do not closely resemble the compositions of genuine Mondrians is surely due to the fact that our model has so few parameters. Noll, after all, had to include numerous hand-tuned parameters in his generative model before any of his faux Mondrians resembled genuine works.³ Our generated images clearly have too many unbroken lines, i.e., lines that span the full canvas either vertically or horizontally to resemble Mondrian’s works.

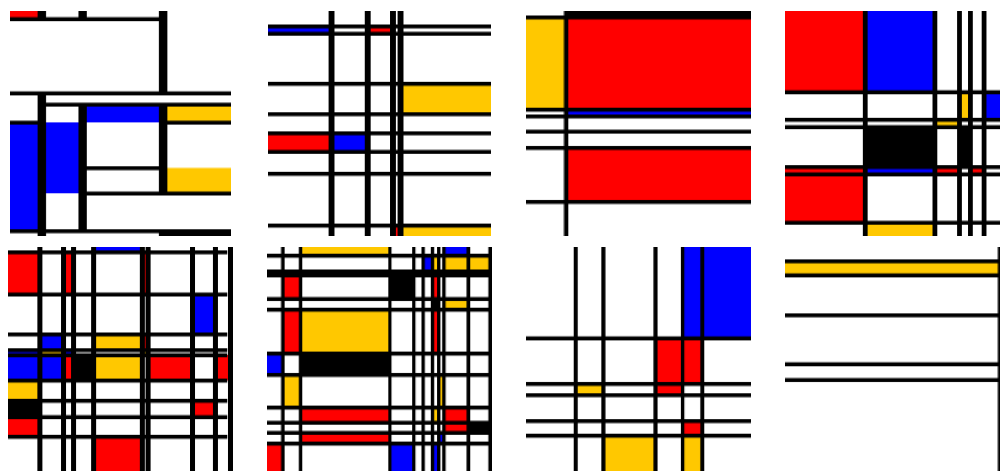


Figure 6. Clearly these examples have a larger number of lines than most works in the training set, and far fewer number of non-spanning lines, that is, lines that terminate at perpendicular lines rather than the outer edge of the painting.

6. CONCLUSIONS

The work reported here is preliminary, a stepping stone towards a richer, more full analysis and synthesis of Mondrian-like compositions. Our automatic image processing and feature extraction methods are adequate for such work, and our compressed representation is adequate to represent genuine Mondrians and intermediate state works alike. We have demonstrated that standard classification methods, trained using leave-one-out cross validation can perform better than chance. Our color weights, extracted from a Mondrian corpus using linear

programming, seem plausible, though validation would require some measure of human psychological response. Our generative model, while surely not as sophisticated as the mind of the artist himself, nevertheless seem powerful enough to represent genuine and “near miss” Mondrians.

As such, there is much research yet to be done. As mentioned in Sect. 2, our models are somewhat preliminary and do not include subtleties of color, brush strokes, and texture. We need to refine our generative model as well. These and related tasks are the subject of ongoing work.

REFERENCES

1. C. Blotkamp, *Mondrian: The art of destruction*, Reaktion Books, London, UK, 1994.
2. T. W. Knight, “Transformations of De Stijl art: The paintings of Georges Vantongerloo and Fritz Glarner,” *Environment and Planning B: Planning and design* **16**, pp. 51–98, 1989.
3. A. M. Noll, “Human or machine: A subjective comparison of a Piet Mondrian’s ‘Composition with lines’ and a computer-generated picture,” *The Psychological record* **16**(1), pp. 1–10, 1961.
4. J. L. Kirsch and R. A. Kirsch, “The anatomy of painting style: Description with computer rules,” *Leonardo* **21**(4), pp. 437–444, 1988.
5. H. Cooper and R. Spronk, eds., *Mondrian: The transAtlantic paintings*, Harvard University Press, Cambridge, MA, 2001.
6. R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern classification*, John Wiley and Sons, New York, NY, Second ed., 2001.
7. D. G. Stork and J. Coddington, eds., *Computer image analysis in the study of art*, vol. 6810, SPIE/IS&T, Bellingham, WA, 2008.
8. D. G. Stork, J. Coddington, and A. Bentkowska-Kafel, eds., *Computer vision and image analysis in the study of art*, vol. 7531, SPIE/IS&T, Bellingham, WA, 2010.
9. S. Lyu, D. Rockmore, and H. Farid, “A digital technique for art authentication,” *Proceedings of the National Academy of Sciences* **101**(49), pp. 17006–17010, 2004.
10. M. Irfan and D. G. Stork, “Multiple visual features for the computer authentication of Jackson Pollock’s drip paintings: Beyond box-counting and fractals,” in *SPIE Electronic Imaging: Image processing: Machine vision applications II*, K. S. Niel and D. Fofi, eds., **7251**, pp. 72510Q1–11, SPIE/IS&T, Bellingham, WA, 2009.
11. C. R. Johnson, E. Hendriks, I. J. Bereznoy, E. Brevdo, S. M. Hughes, I. Daubechies, J. Li, E. Postma, and J. Z. Wang, “Image processing for artist identification,” *IEEE Signal Processing magazine* **25**(4), pp. 37–48, 2008.
12. D. G. Stork, “Optics and realism in Renaissance art,” *Scientific American* **291**(6), pp. 76–84, 2004.
13. R. C. Gonzalez and R. E. Woods, *Digital image processing*, Prentice Hall, Upper Saddle River, NJ, Third ed., 2007.
14. L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *CART: Classification and regression trees*, Chapman and Hall, New York, NY, revised ed., 1993.
15. I. Grebert, D. G. Stork, R. Keesing, and S. Mims, “Network generalization for production: Learning and producing styled letterforms,” in *Advances in neural information processing systems*, J. E. Moody, S. J. Hanson, and R. P. Lippmann, eds., **4**, pp. 1118–1124, Morgan Kaufmann Publishers, (San Mateo, CA), 1992.