FORSCHUNGSBERICHTE
KÜNSTLICHE INTELLIGENZ

Algorithmic Art

Jürgen Schmidhuber

Report FKI-197-94
September 1994

TUM

TECHNISCHE UNIVERSITÄT MÜNCHEN
Institut für Informatik (H2), D-80290 München, Germany

ISSN 0941-6358

Forschungsberichte Künstliche Intelligenz

ISSN 0941-6358

Institut für Informatik
Technische Universität München

Die Forschungsberichte Künstliche Intelligenz enthalten vornehmlich Vorab-Veröffentlichungen, spezialisierte Einzelergebnisse und ergänzende Materialien, die seit 1988 in der KI / Kognitionsgruppe am Lehrstuhl Prof. Brauer bzw. 1988-1993 in der KI / Intellektik Gruppe am Lehrstuhl Prof. Jessen entstanden. Im Interesse einer späteren Veröffentlichung wird gebeten, die Forschungsberichte nicht zu vervielfältigen. Alle Rechte und die Verantwortung für den Inhalt des Berichts liegen bei den Autoren, die für kritische Hinweise dankbar sind.

Eine Zusammenstellung aller derzeit lieferbaren FKI-Berichte und einzelne Exemplare aus dieser Reihe können Sie bei folgender Adresse anfordern oder über ftp beziehen:

The "Forschungsberichte Künstliche Intelligenz" series includes primarily preliminary publications, specialized partial results, and supplementary material, written by the members of the AI / Cognition Group at the chair of Prof. Brauer (since 1988) as well as the "Intellektik" Group at the chair of Prof. Jessen (1988-1993). In the interest of a subsequent final publication these reports should not be copied. All rights and the responsibility for the contents of the report are with the authors, who would appreciate critical comments.

You can obtain a list of all available FKI-reports as well as specific papers by writing to the adress below or via ftp:

"FKI"
Institut für Informatik (H2)
Technische Universität München
D-80290 München
Germany

Phone: +49 - 89 - 2105 2406
Telex: tumue d 05-22854
Fax: +49 - 89 - 2105 - 8207
e-mail: fki@informatik.tu-muenchen.de

FTP:
machine: flop.informatik.tu-muenchen.de
or 131.159.8.35
login: anonymous
directory: pub/fki

ALGORITHMIC ART
Report FKI-197-94

Jürgen Schmidhuber
Fakultät für Informatik
Technische Universität München
80290 München, Germany

schmidhu@informatik.tu-muenchen.de
http://papa.informatik.tu-muenchen.de/mitarbeiter/schmidhu.html

September 1994

Abstract

Many artists try to depict "the essence" of objects to be represented. In an attempt to formalize certain aspects of the "the essence", I propose an art form called *algorithmic art*. Its goals are based on concepts from algorithmic information theory. Suppose the task is to draw a given object. Usually there are many ways of doing so. The goal of algorithmic art is to draw the object such that the drawing can be specified by a computer algorithm and two properties hold: (1) The drawing should "look right". (2) The Kolmogorov complexity of the drawing should be small (the algorithm should be short), and a typical observer should be able to see this. Examples of algorithmic art are given in form of "algorithmically simple" cartoons of various objects, including a pin-up girl and a weight lifter. Relations to previous work are established. Attempts are made to relate the formalism of the theory of minimum description length to informal notions like "good artistic style" and "beauty".

Keywords: *Algorithmic art, fine arts, Kolmogorov complexity, algorithmic information, beauty, attractiveness, circles, fractals.*

Disclaimer: Despite being published in a tech report series, this is *not* a scientific paper. Although attempts are made to formalize certain aspects of informal notions like "art" and "beauty", much of what is said remains informal and speculative.

 Note: There is a long version of this paper (Schmidhuber, 1994a). It has 68 pages, contains 27 figures, and is written in German.

1 INTRODUCTION

In their introduction to Kolmogorov complexity, Li and Vitányi (1993) write:

> "We are to admit no more causes of natural things (as we are told by Newton) than such as are both true and sufficient to explain their appearances. This central theme is basic to the pursuit of science, and goes back to the principle known as Occam's razor: 'if presented with a choice between indifferent alternatives, then one ought to select the simplest one'. Unconsciously or explicitly, informal application of this principle in science and mathematics abound."

This paper argues that the principle of Occam's razor is not only relevant to science and mathematics, but to fine arts as well. Some artists consciously prefer "simple" art by claiming: "art is the art of omission". Furthermore, many famous works of art were either consciously or unconsciously designed to exhibit regularities that intuitively simplify them. For instance, every stylistic repetition and every symmetry in a painting allows for describing one part of the painting in terms of descriptions of other parts. Intuitively, redundancy of this kind helps to shorten the length of the description of the whole painting, thus making it "simple" in a certain sense.

It is possible to *formalize* the intuitive notions of "simplicity" and "complexity". The appropriate mathematical tools are provided by the theory of Kolmogorov complexity or algorithmic complexity (Kolmogorov, 1965; Chaitin, 1969; Solomonoff, 1964). See (Li and Vitányi, 1993) for the best overview. See (Schmidhuber, 1994b) for a machine learning application. The Kolmogorov complexity of a computable object is defined as the length of the shortest program for a universal computer (or Turing machine) that computes the object. Kolmogorov complexity is "objective" in the sense that it is essentially independent of the particular computer used, leaving aside an additive machine specific constant. This fact is known as the invariance theorem (Solomonoff, 1964; Kolmogorov, 1965; Chaitin, 1969). The reason for the invariance theorem is that any program for a given machine can be compiled into an equivalent program for a given universal machine by a compiler program of constant size.

In this paper, basic concepts from the theory of algorithmic complexity serve as ingredients for a novel form of art. Although the focus will be on black-and-white cartoons, the basic ideas are not limited to them.

2 ALGORITHMIC ART

Suppose an artist's task is to produce a drawing which obeys a set of (possibly informal) specifications given in advance. The goal of algorithmic art is to represent "the essence" by achieving two conflicting goals simultaneously:

> **Goal 1.** Given the specifications, the drawing should "look right".
>
> **Goal 2.** (A) The Kolmogorov complexity of the final design should be provably small. In other words, there should be a short algorithm computing the drawing (by generating appropriate instructions for a printer, say). (B) It should be easy for the observer to perceive the algorithmic simplicity of the drawing. He ought to *see* the "essence" extracted by the algorithmic artist.

It is predicted that drawings that do a good job on both conflicting goals will be appreciated by the observer. The next subsection addresses the extent to which both goals are subjective.

2.1 HOW SUBJECTIVE IS ALGORITHMIC ART?

Goal 1 is clearly subjective in the sense that it strongly depends on a given observer, and on the way he interprets the (possibly informal) specifications. What "looks right" to an observer from one (sub)culture may "look wrong" to an observer from another (sub)culture (or another time).

Goal 2(A) depends on the nature of the computer running the algorithm. In what follows, this dependency will be ignored. In the limit, this is justified by the above-mentioned invariance theorem (Solomonoff, 1964; Kolmogorov, 1965; Chaitin, 1969).

Like Goal 1, Goal 2(B) depends on the observer. But in a sense, Goal 2(B) is less subjective than Goal 1. This is because intelligent human observers in principle can *learn* to compute anything a digital computer can compute (only the converse is a matter of controversy). In particular, a short algorithm running on a conventional digital machine can be quickly taught to an intelligent human being. Note that if the human observer was just another universal computer, then we could immediately apply the invariance theorem, thus (in the limit) removing subjectiveness from Goal 2(B). Then the only remaining subjective aspect of algorithmic art would be given by Goal 1.

2.2 Algorithmic Art is Hard

This paper describes the goals of algorithmic art, not the way to achieve the goals. The latter requires intuition and, like with any other form of art, a sometimes rewarding but often frustrating struggle for capturing "the essence". In the beginning of his attempts to create a work of algorithmic art, the algorithmic artist will usually not be able to predict the details of the final result.

A fundamental theorem from algorithmic information theory says that there is no general method for finding the shortest description of some piece of data (Kolmogorov, 1965; Solomonoff, 1964; Chaitin, 1969). This seems to indicate that the formal art form proposed above will always represent a big challenge to any artist willing to pursue it.

2.3 Algorithmic Design

To have regard for the difference between art and design (this difference is important to many artists), I would like to use the expression *"algorithmic design"* instead of *"algorithmic art"* in cases where no artistic purpose is pursued by a designer trying to achieve Goal 1 and Goal 2.

2.4 Outline of Remainder of Paper

Section 3 first defines a "fractal" (Mandelbrot, 1982) coding scheme for encoding drawings in algorithmic form. The coding scheme is easily learned and understood by most humans. The bulk of this paper is devoted to examples of algorithmic art. Section 4 presents a set of cartoons. Each cartoon satisfies some informal specifications provided in advance. Furthermore, each cartoon is "algorithmically simple" – its description (based on the coding scheme implemented on a conventional digital computer) does not require many bits of information. To achieve Goal 2(B), the algorithmic simplicity of each cartoon is made obvious with the help of text and additional drawings. Section 5 attempts to relate the formalism of the theory of minimum description length (MDL) to informal notions like "good artistic style" and "beauty". Section 6 establishes relations to previous work and concludes with an outlook on the possible future of algorithmic art.

3 Circles for Coding Drawings

This section introduces a "fractal" coding scheme for encoding drawings in algorithmic form. The scheme is general enough to design arbitrary drawings. As will be seen in section 4, it is sophisticated enough to allow for specification of non-trivial drawings with a very limited amount of information. Finally, it is "simple" enough to be implemented by a short algorithm, and to be quickly taught to typical human observers.

The ancient Greeks considered the circle as the "ideal" two-dimensional geometric form. Without necessarily agreeing with the Greeks, in what follows I will use circles as the basis for designing drawings. One motivation is that a circle can be drawn by a very short algorithm. Another motivation is that circles are something most humans can relate to: most people know something about circles and their properties. This will make it easy to teach the algorithmic simplicity of the drawings from section 4 to a typical observer (to achieve Goal 2(B) from section 2).

Sizes and relative positions of "legal" circles will be greatly limited by the following "fractal" rules.

3.1 Rules for Making Legal Circles

> **Initialization:** *Arbitrarily define the first circle. Arbitrarily select a second circle with equal radius whose center is on the first circle. The first two circles are defined to be "legal" circles.*

The rules for generating additional legal circles are as follows:

> **Rule I.** *Wherever two legal circles of equal radius touch or intersect, another legal circle of equal radius may be drawn with this point as its center.*
>
> **Rule II.** *Every legal circle with center point p and radius r may have within it a legal circle whose center point is also p but whose radius is r/2.*

Figure 1 shows the result of an iterative application of the rules above.

3.2 Rules for Making Legal Drawings

A *legal drawing* is defined by (a) *legal lines* (segments of legal circles) or (b) *legal areas* (defined by legal circles that touch or intersect). The rules for legal lines and areas are:

> **Rule III.** *Each legal line must be a segment of a legal circle.*
>
> **Rule IV.** *At both endpoints of a legal line some legal circles must touch or intersect.*
>
> **Rule V.** *The line width of a legal line must be equal to the radius of some legal circle.*
>
> **Rule VI.** *A legal area is an area whose border is a closed chain of legal lines. Legal areas may be shaded using a small set of grey levels.*

Comments

(1) On each legal circle you can find the centers of 6 legal circles with the same radius.
(2) Define the radius of the initial circle as 1. The radius of any legal circle can be written as 2^{-n}, where n is a non-negative integer. The same holds for legal line widths.
(3) On a given area, there are about 4 times as many circles with radius 2^{-n-1}, as there are circles with radius 2^{-n}.
(4) Algorithmic art is in no way limited to Rules I-VI. For instance, a drawing that only partly obeys Rules I-VI may be a work of algorithmic art, as long as the deviations can be uniquely determined by a short algorithm.

3.3 Coding Drawings by Circle Numbers

There are many straight-forward schemes for encoding drawings generated by rules I-VI. This section gives an example.

Have another look at Figure 1. Define the radius of the initial circle (the frame) as 1. A *visible circle* is a circle partly covered by the initial circle. Starting with the initial circle, we generate all visible circles. Each gets a number. The initial circle gets the number 1. There are 12 visible circles with radius 1 intersecting the initial circle. They are numbered 2, 3, ..., 13 (in some deterministic clockwise fashion, say). There are 31 visible circles with radius $\frac{1}{2}$ (partly) covered by the initial circle. They are numbered 14, 15, ..., 44. And so on.

Obviously, there are few big circles with small numbers. There are many small circles with high numbers. In general, the smaller a circle, the more bits are needed to specify its number.

An unshaded drawing is specified by a set of legal lines. For each legal line l, we need to specify the number of the corresponding circle c_l, the start point s_l, the end point e_l, and the line width w_l. By convention, lines are drawn clockwise from s_l to e_l. Once we know c_l, we can specify s_l by specifying the number of the circle touching or intersecting c_l in s_l. In general, an extra bit is necessary to disambiguate between two possible intersections. Similarily for e_l. Thus, all "pixels" of a legal line may be compactly represented by: a triple of circle numbers, two bits for intersection disambiguation, and a few bits for the line width.

Clearly, the bigger the used circles, the fewer bits are needed to specify the corresponding legal lines, and the simpler (in general) the drawing. By using very many very small circles (beyond the resolution of the human eye), anything can be drawn (using Rules IV and V) such that it "looks right". This would not be very impressive, however, because a lot of information would be required to specify the drawing. It would be more impressive if it were possible to draw something non-trivial that "looks right" from legal lines defined by few big circles. In a way, this would be related to "capturing the essence", provided one agrees that the "essence" of an object rests in the shortest algorithm describing the object. Section 6, however, will show that such compact representation can be difficult. I found that it is much easier to come up with acceptable complex drawings than with acceptable simple drawings of given objects.

Often the algorithmic artist will use drawing-specific symmetries and the like to further compress the description of a drawing. The next section will present examples of this.

4 Examples of Cartoons Based on Rules I-IV

Throughout the centuries, most artist's favorite object has been the human body. In what follows, I will for the most part adhere to this tradition. Here is a set of informal specifications of the cartoons to be presented:

Figure 2: The (informal) goal is to design a cartoon of a girl's face for a comic book.

Figure 4: The goal is to draw a cartoon of a pin-up girl with the face from Figure 2.

Figure 8: The goal is to design a logo for a gym based on a weight lifter's upper body.

Figure 10: The goal is to draw 10 flowers with 6 leaves each, for J. Sickinger's 60th birthday[1].

Figures 2, 4, 6, and 8 are examples of cartoons designed by using rules I-VI only. Instead of providing each drawing's somewhat opaque coding sequence (a list of small numbers generated according to the coding scheme from section 3.3), Figures 3, 5, 7, and 9 provide corresponding graphical "explanations". In conjunction with Figure 1, each explanation allows the algorithmic simplicity of the corresponding cartoon to be described quickly to the human observer.

The figures demonstrate that the circle scheme is quite flexible. In terms of bits, it is cheaper to encode all cartoons (Figures 2, 4, 6, 8) simultaneously than to encode each cartoon separately because the algorithm for generating legal circles and their numbers (see section 3.3) is shared by all four cartoons. In the terminology of algorithmic information theory, the cartoons share a non-trivial amount of mutual algorithmic information. The circle scheme may be viewed as something like a common recognizeable "artistic style".

Many additional examples of algorithmically simple drawings, including those I consider to be my most interesting, appear elsewhere (Schmidhuber, 1994a).

5 On Beauty and Minimum Description Length

Sometimes, an artist is appreciated for his distinctive style. Sometimes, certain works of art are perceived as "beautiful". This section attempts to relate the formalism of the theory of minimum description length

[1] J. Sickinger is the author's uncle.

(MDL, see (Solomonoff, 1964; Kolmogorov, 1965; Chaitin, 1969; Levin, 1974; Wallace and Boulton, 1968; Rissanen, 1978) for important contributions, see (Li and Vitányi, 1993) for an overview) to informal notions like "beauty" and "good artistic style".

5.1 WHAT IS A BEAUTIFUL DRAWING?

What is beautiful? What is not? There can clearly be no objective answers to these questions. What is considered beautiful by one observer may be regarded ugly by another observer. Ideals of beauty are different in different cultures and subcultures; they have changed over the centuries; and they are not even stable with respect to a single individual. Therefore, any "theory of beauty" has to take the observer (the subject) into account.

Following common intuition, I assume that a typical human observer tries to represent input data in terms of what he already knows. To take care of the observer's subjectiveness, I assume that the Church-Turing thesis[2] is true and postulate the following setting. At a given time, the current knowledge of a human observer can be described as a coding algorithm. This algorithm maps input data (such as retinal activation caused by a work of art in the visual field) onto "internal representations" of the data. The coding algorithm C, the data D, and its internal representation D', can be written as strings of symbols from a finite alphabet. If D' conveys all the information about D, but the length of D' is less than the length of D, then D is compressible or redundant with respect to the observer's knowledge. The observer already "knew something about D". Similar statements can be made in cases where D' allows only for partial reconstruction of D.

The observer's subjectiveness is embodied by his coding algorithm C. One may be tempted to define the "beauty" of a drawing with respect to C. In the following preliminary attempt to do so (inspired by the MDL approach), I assume that "beauty" simply corresponds to "high probability with respect to C": Given C, the "best" way of selecting a drawing s from a set or class S of possible drawings satisfying certain specifications may be to maximize $P(s \mid C)$, the conditional probability of s, given C. Bayes' formula tells us

$$P(s \mid C) = \frac{P(C \mid s)P(s)}{P(C)}, \tag{1}$$

or, equivalently,

$$-\,logP(s \mid C) = -logP(C \mid s) + logP(C) - logP(s). \tag{2}$$

Let us interpret this. Since C is given, $P(C)$ may be viewed as a normalizing constant. It may be neglected. $-logP(C \mid s)$ can be interpreted as the information (or length of the observer's shortest algorithm) required to compute C from s. $P(s)$ is given by some *a priori* distribution on the drawings. For simplicity, let us assume that this prior is uniform. Then, given C, $s \in S$ is optimal (most likely, most "beautiful") if the information required to compute C from s is minimized.

How can this be related to human experience? The following example attempts to establish such a relationship.

"Beautiful" faces. Humans appear to have a certain coding scheme for storing faces. This scheme is certainly different from the circle scheme from section 3. It is probably based on previous experiences with many different faces, and it is probably adapted to code many faces efficiently (see also section 5.2). One way of doing so is to store a prototype face and code new faces by coding only the deviations from the prototype.

The principle of minmal description length suggests that the "ideal" (most likely) prototype F_P maximizes $P(F_P \mid F)$, thus minimizing

$$-logP(F \mid F_P) - logP(F_P),$$

where F is a given set of all faces to be coded. In words: The "optimal" prototype minimizes the sum of the description lengths of all faces relative to the prototype, and the description length of the prototype itself (relative to the remaining observer knowledge).

[2]Loosely speaking, the Church-Turing thesis claims: everything that can be computed by a human being can be computed by an appropriate program for a general purpose computer (like a Turing machine).

Assume that all faces are equally likely to appear in the visual field. The formalism above predicts that the most beautiful face is the one that makes it easiest to compute the coding scheme from the face. It seems reasonable to assume that the information required to specify the coding scheme is dominated by the information required to specify the prototype face. If the current face looks like the prototype face, then there is hardly anything to compute. This would imply that the prototype face is perceived as the most beautiful one.

This seems compatible with results presented in (Langlois and Roggman, 1990). The authors claim that the "average face" (computed by digital blending of numerous photos of real faces) is perceived as the most attractive one. Perrett, May and Yoshikawa (1994) partly dispute this claim, however. Their test subjects also appreciate "average faces" computed by blending (Benson and Perrett, 1993) but prefer "attractive average faces" constructed from faces perceived as attractive. Indeed, the most attractive faces are "caricatures" obtained by digitally exaggerating the deviations between "average" and "attractive average".

The studies above, however, do not say much about the plausibility of the algorithms used to compute "average faces". Let us assume that the brain does indeed support face processing by an "ideal" (in the information theoretic sense) prototype face. It would be naive to assume that the latter equals the one computed by blending. There are many possible plausible algorithms for computing prototypes, based on many plausible metrics for "distances" between faces. Therefore the studies above, including the statement that the average face is not the most attractive one, have to be judged with scepticism. The presented claims depend on the definition of "the average" and the corresponding nature of the blending algorithms, which need not be very much related to a hypothetical method the brain might be using for generating the "optimal" prototype F_P. Unfortunately, at the present time it seems impossible to analyze the way the brain stores representations of objects. Therefore it also seems impossible to test the predictions made by the formalism presented above.

Comments

1. One may continue to speculate as follows. A society with a distribution of faces corresponding to an algorithmically simple prototype face may have an evolutionary advantage. This is because face recognition (based on the given hardware, the brain) may be more successful or efficient in such a society. Evolutionary pressure may favor "beautiful" prototypes, where "beauty" is defined by the nature of the computations our brain is good at. On the other hand, the nature of these computations is influenced by typical face recognition tasks to be solved. It seems to be hard to analyze such mutual dependencies.

2. Most certainly, the circle scheme from section 3 is different from the typical human coding scheme. Therefore, the "most beautiful" cartoons relative to the circle scheme will be different from the "most beautiful" cartoons relative to the coding scheme of most humans. Unfortunately, I cannot extract the latter (although many artists implicitly try to guess it, I believe). However, humans can learn new coding schemes. In particular, it is not hard for them to learn the circle scheme. Therefore I hope that some of the cartoons in this paper will be easily accessible by some readers.

3. Something "beautiful" needs not be "interesting". "Interestingness" has to do with "unexpectedness". But not everything that is unexpected is interesting – just think of "white noise". One reason for the "interestingness" (relative to some observers) of some of the pictures from section 4 may be that they exhibit "unexpected structure". Certain aspects of these pictures are not only unexpected (relative to typical observers), but unexpected in a regular, non-random way. The formalization of "interestingness" requires an extension of the formalism above. This, however, is beyond the scope of the paper.

5.2 What is Good Artistic Style?

Above, I made use of the fact that humans can learn new coding algorithms. In a way, their original coding algorithm is universal enough to allow for implementing new coding algorithms. One important subgoal of algorithmic art is to devise "good" coding algorithms. What does this mean? A new coding scheme may be considered as "good" relative to a given observer if (1) it does not require many bits to be specified (given the observer's previous knowledge) and (2) many different drawings (satisfying typical specifications) can be encoded efficiently by it. In that case the drawings share a non-trivial amount of algorithmic information, and the coding scheme represents something like a common style. A given coding scheme may be representative for a given artist, which will make him stylistically recognizeable.

More formally, the quality of an artistic style or coding scheme C may be evaluated as follows. An "optimal" style C maximizes $P(C \mid S)$, the conditional probability of the style, given a set of drawings S (defined by a set of specifications). Equivalently, an "optimal" style C minimizes

$$- logP(C \mid S) = -logP(S \mid C) + logP(S) - logP(C). \tag{3}$$

Since S is given, $P(S)$ may be viewed as a normalizing constant, and it may be ignored. The term, $-logP(S \mid C)$, can be interpreted as the information required to compute all elements in S from C. The term, $P(C)$, is given by some *a priori* distribution on the coding schemes and depends on the observer. $-logP(C)$ can be interpreted as the information necessary to specify C, given the observer's knowledge. Thus, given S, C is optimal (most likely) if the sum of two terms is minimized: (1) The information required to compute S from C, and (2) The information required to compute C from the observer's previous coding scheme.

The circle scheme from section 3 is easy to teach to typical humans, which may be another way of saying that not much information is required to compute it from typical human knowledge. In this case, the second term appears negligible. Therefore, given the drawings presented in this paper, the circle scheme appears to correspond to a "good" (although probably non-optimal) artistic style.

6 Final Remarks

6.1 On the Difficulty of Creating Algorithmic Art

This paper specifies (and exemplifies) the nature of algorithmic art without providing a general method for *creating* algorithmic art. Although it is trivial to redraw the concrete examples from section 4, they appear "out of the blue". They do not tell how they were discovered. They do not give many clues about how to draw other objects. No universal algorithm for generating algorithmic art is known. A human artist is required.

I found it difficult to discover acceptable but algorithmically simple cartoons. I found it easier to come up with acceptable cartoons that appeared to be algorithmically very complex. For instance, I had to do more than 2000 sketches, all of them seriously flawed in one way or another, before finally ending up with the algorithmically simple Figure 4.

6.2 Relation to Previous Work

This paper certainly is not the first to introduce formal rules in art. For instance, Lyonel Feininger writes:

> ***"Aber die Erkenntnis ist mein, dass es in der ganzen Welt, in allen Welten, nichts Ungesetzliches, nichts Zufälliges, nichts ohne Form und Rhythmus gibt noch geben kann. Warum dann gerade in der Kunst? Soll diese nicht dann, indem sie des Menschen schöpferischen Willen offenbart, gerade voll Form, voll Gesetz und Geist sein?"***
>
> English translation[3]: ***"But the insight is mine, that in the whole world, in all worlds, nothing unlawful, nothing random, nothing without form and rhythm exists nor can exist. Why then in the arts? Shouldn't the arts, by exhibiting man's creative will, be filled with form, law and spirit?"***

[3]Free translation by the author. Subtleties in the original text source may be lost.

The ancient Greeks, Leonardo da Vinci, Albrecht Dürer, LeCorbusier and many others devised formal rules to "draw things right". Most rules are based on "simple" proportions. Examples are: "The distance between the eyes should equal the eye-width" (origin unknown). Or: "The ratio of the distance from toes to navel and the distance from toes to top of the head should be the harmonic proportion[4]" (LeCorbusier).

Artists know many rules like that. Most of these rules, however, are very informal and leave almost everything to intuition. In contrast to previous informal approaches, this paper adopts an extreme standpoint. It advocates the position that the simplicity of a work of art should be *proven* by demonstrating that it can be computed by a short algorithm. Nothing in a work of algorithmic art should be without such a motivation.

This paper does not claim to be the first to present examples of algorithmic art, however. Certain representations of artistically interesting and aesthetically pleasing representations of "fractal" objects may be regarded as works of algorithmic art, such as hills, coast lines etc. (Mandelbrot, 1982; Peitgen and Richter, 1986). This is because they are based on relatively short and understandable algorithmic descriptions. Also, certain data compression methods can be used to generate patterns that may be regarded as works of algorithmic art (see e.g. (Culik and Kari, 1994) and the references to be found therein). Similar statements could be made about certain other simplifying computer models for representing objects (see e.g. (Essa et al., 1994) for work on face animation). See also (Li, 1992). Also, many graphical designers consciously or unconsciously use their tools to come up with algorithmically simple designs.

Apparently, however, nobody so far has identified algorithmic art and algorithmic design as such in its most general form. A contribution of this paper is to make explicit the nature of algorithmic art and algorithmic design – the creation of understandable works of art (or designs) with low Kolmogorov complexity. This also provides a framework for the categorization of previous work.

Finally it should be mentioned that there have been attempts to use classical information theory (Shannon, 1948) to formalize what's aesthetically pleasing, e.g. (Nake, 1974). Another contribution of this paper is to offer an alternative approach based on *algorithmic* information theory.

6.3 Outlook

Artistic expressibility depends on available technology. The cave artists from the stone age did not have the technology for creating colors that made impressionism possible. The impressionists did not have today's computer graphics. This does neither imply that the best cave drawings are less perfect than the best impressionist paintings, nor that the best impressionist paintings are less perfect than the best computer graphics. Each age, however, tends to have its preferred means of artistic expression. What can we expect with regard to the future of algorithmic art?

There will be tools that will simplify the creation of algorithmic art. I expect significant extensions of the programs used to speed up and simplify the creation of the drawings from section 4. In particular, I expect "virtual ateliers" implemented on powerful machines. A virtual atelier will be accessible via stereoscopic virtual reality interfaces. It will allow complex 3-dimensional (or more dimensional) objects to be quickly composed from simpler ones by hand movements (perceived by the machine via data gloves or similar devices). For instance, with a virtual atelier it will be easy to extend the circle scheme from section 3 to an analoguous "sphere scheme" or "bubble scheme". The algorithmic atelier will permit the artist to quickly generate sequences of 3-dimensional "sketches" of sculptures, to evaluate them with respect to their artistic value, to discard them or refine them.

In principle, the technology for building virtual ateliers is available. Given the current inflation of cheap computing power, we may expect that it will not take very long until many artists will have access to acceptable virtual ateliers.

[4] The harmonic proportion is what you get if you divide a straight line of length a into two segments of lenghts b and c, such that $\frac{a}{b} = \frac{b}{c}$. One solution is $b = a\frac{\sqrt{5}-1}{2}$.

7 ACKNOWLEDGEMENTS

Thanks to Ulrike Krommer, Christian Freksa, Mark Ring, Christof Schmidhuber, Kurt Prelinger, Karolin Schmidhuber, Gerhard Weiss, Astrid Kiehn, Rohan Baxter, Margit Kinder, Martin Eldracher, Daniel Prelinger, and Sepp Hochreiter, for useful comments.

References

Benson, P. J. and Perrett, D. I. (1993). Extracting prototypical facial images from exemplars. *Perception*, 22:257–262.

Chaitin, G. (1969). On the length of programs for computing finite binary sequences: statistical considerations. *Journal of the ACM*, 16:145–159.

Culik, K. and Kari, J. (1994). Parallel pattern generation with one-way communications. In Karhumäki, J., Maurer, H., and Rozenberg, G., editors, *Results and trends in Theoretical Computer Science*, pages 85–96. Springer.

Essa, I. A., Darrell, T., and Pentland, A. (1994). Modeling and interactive animation of facial expressions using vision. Technical report, MIT Media Lab Perceptual Computing Section TR 256.

Kolmogorov, A. (1965). Three approaches to the quantitative definition of information. *Problems of Information Transmission*, 1:1–11.

Langlois, J. H. and Roggman, L. A. (1990). Attractive faces are only average. *Psychological Science*, 1:115–121.

Levin, L. A. (1974). Laws of information (nongrowth) and aspects of the foundation of probability theory. *Problems of Information Transmission*, 10(3):206–210.

Li, M. (1992). Minimum description length based 2-d shape description. In *IEEE 4th Int. Conf. on Computer Vision*, pages 512–517.

Li, M. and Vitányi, P. M. B. (1993). *An Introduction to Kolmogorov Complexity and its Applications*. Springer.

Mandelbrot, B. (1982). *The Fractal Geometry of Nature*. Freeman and Co., San Francisco.

Nake, F. (1974). *Ästhetik als Informationsverarbeitung*. Springer.

Peitgen, H. O. and Richter, P. H. (1986). *The Beauty of Fractals. Images of Complex Dynamical Systems.* Springer.

Perrett, D. I., May, K. . A., and Yoshikawa, S. (1994). Facial shape and judgements of female attractiveness. *Nature*, 368:239–242.

Rissanen, J. (1978). Modeling by shortest data description. *Automatica*, 14:465–471.

Schmidhuber, J. H. (1994a). Algorithmische Kunst. Manuscript.

Schmidhuber, J. H. (1994b). Discovering problem solutions with low Kolmogorov complexity and high generalization capability. Technical Report FKI-194-94, Fakultät für Informatik, Technische Universität München.

Shannon, C. E. (1948). A mathematical theory of communication (parts I and II). *Bell System Technical Journal*, XXVII:379–423.

Solomonoff, R. (1964). A formal theory of inductive inference. Part I. *Information and Control*, 7:1–22.

Wallace, C. S. and Boulton, D. M. (1968). An information theoretic measure for classification. *Computer Journal*, 11(2):185–194.

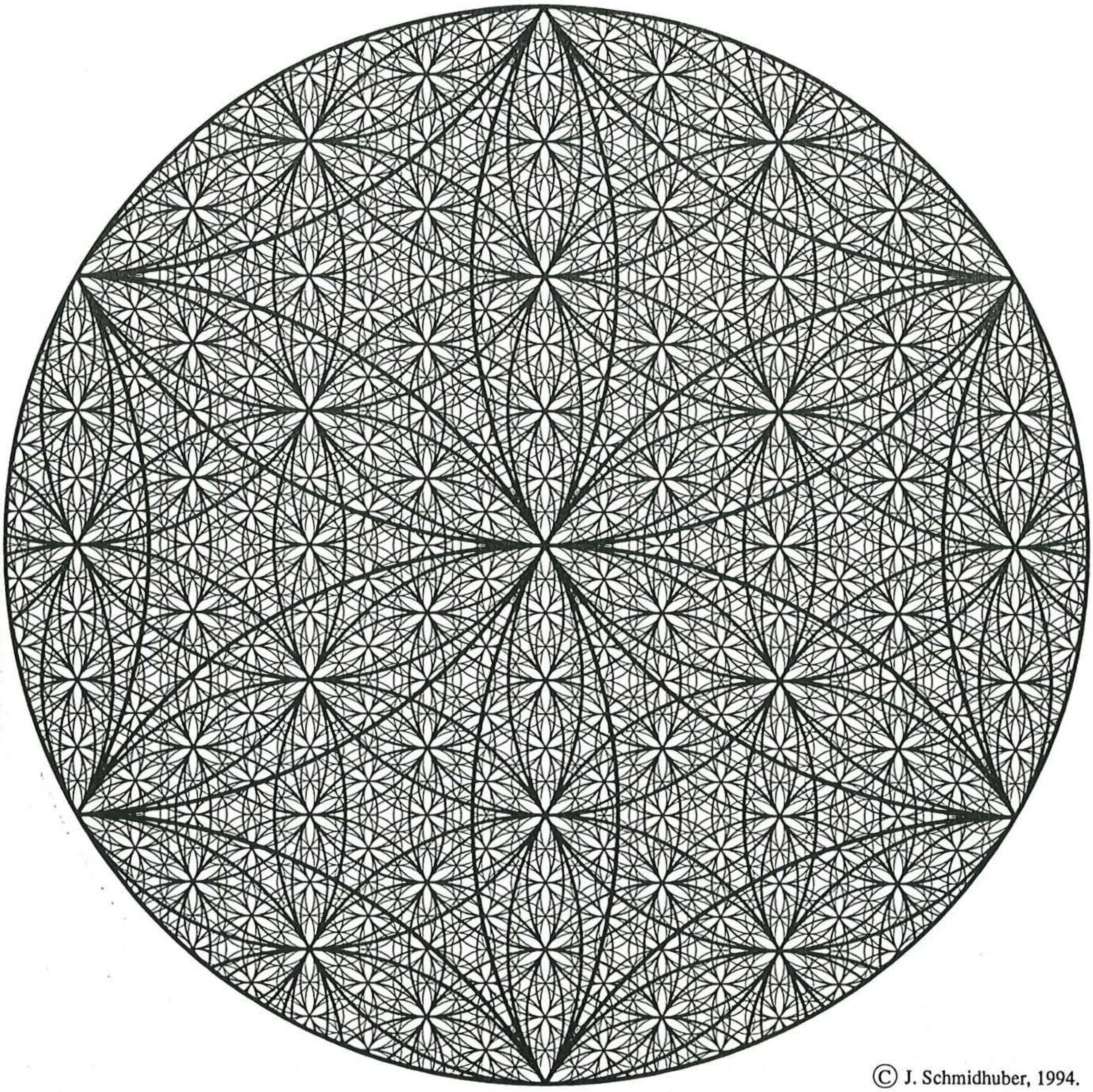

Figure 1: "Fractal" result of an iterative application of Rules I and II (following initialization, see section 3.1). Recall that Rule I says: "wherever two legal circles of equal radius touch or intersect, define the center of another legal circle with equal radius." Recall that Rule II says: "the center of any legal circle with a given radius is the center of another legal circle with half the radius." The above "self-similar" figure builds the basis of all figures to follow. On each legal circle there are the centers of 6 legal circles with equal radius. For clarity, legal circles are emphasized in proportion to the logarithm of their radius. Parts of legal circles located outside the initial circle (the "frame") are not shown. Also, circles whose radius is smaller than one sixteenth of the radius of the frame are not shown.

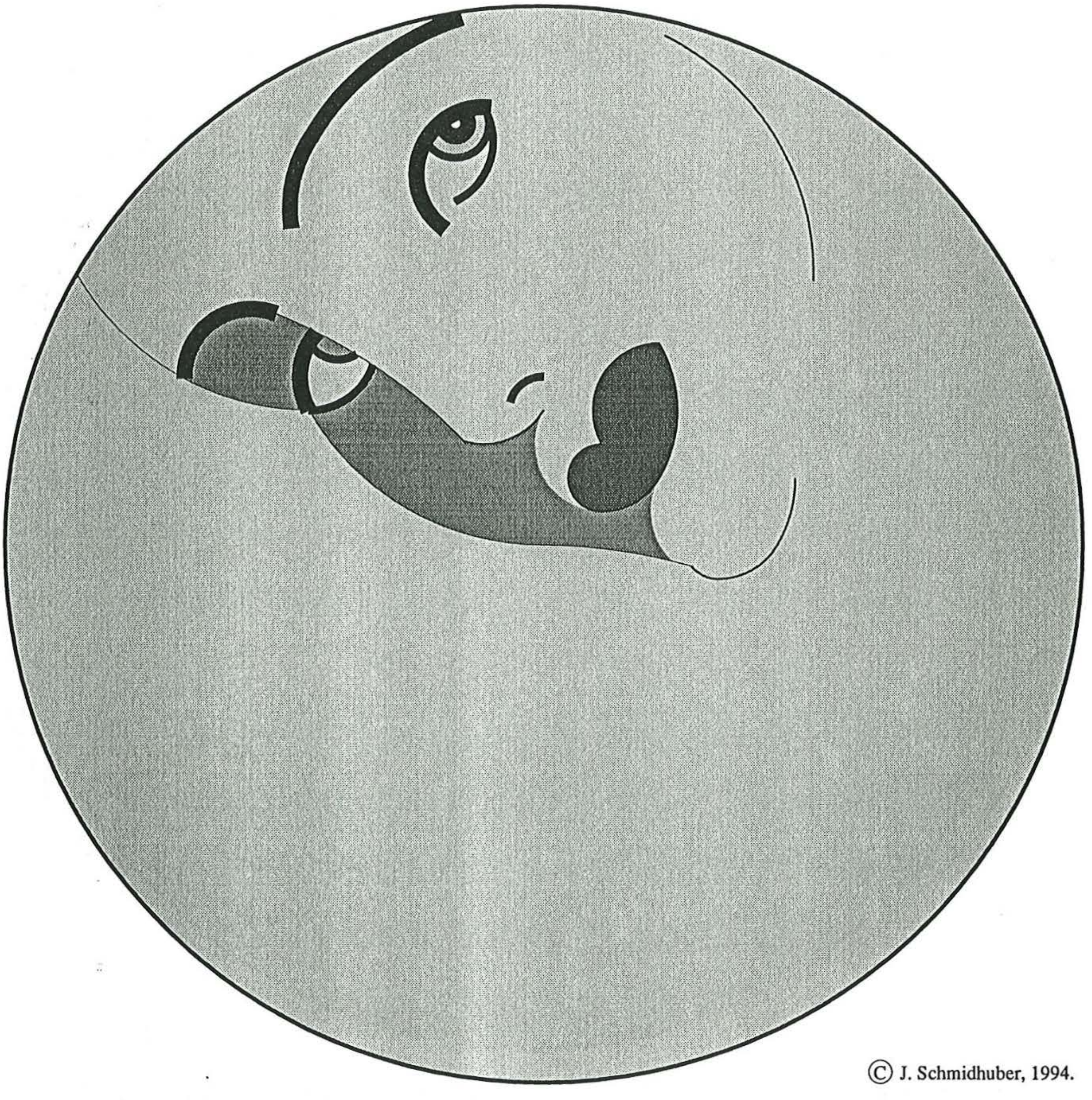

Figure 2: Cartoon of a girl's face for a comic book. Only Rules I-VI (see section 3) were used to design the drawing: all contours of the face are segments of legal circles from Figure 1. The cartoon can be specified by a short sequence of small integers, using the coding scheme from section 3.3. Therefore, the cartoon is "algorithmically simple". Its low Kolmogorov complexity (or its algorithmic simplicity) can be explained by Figure 3.

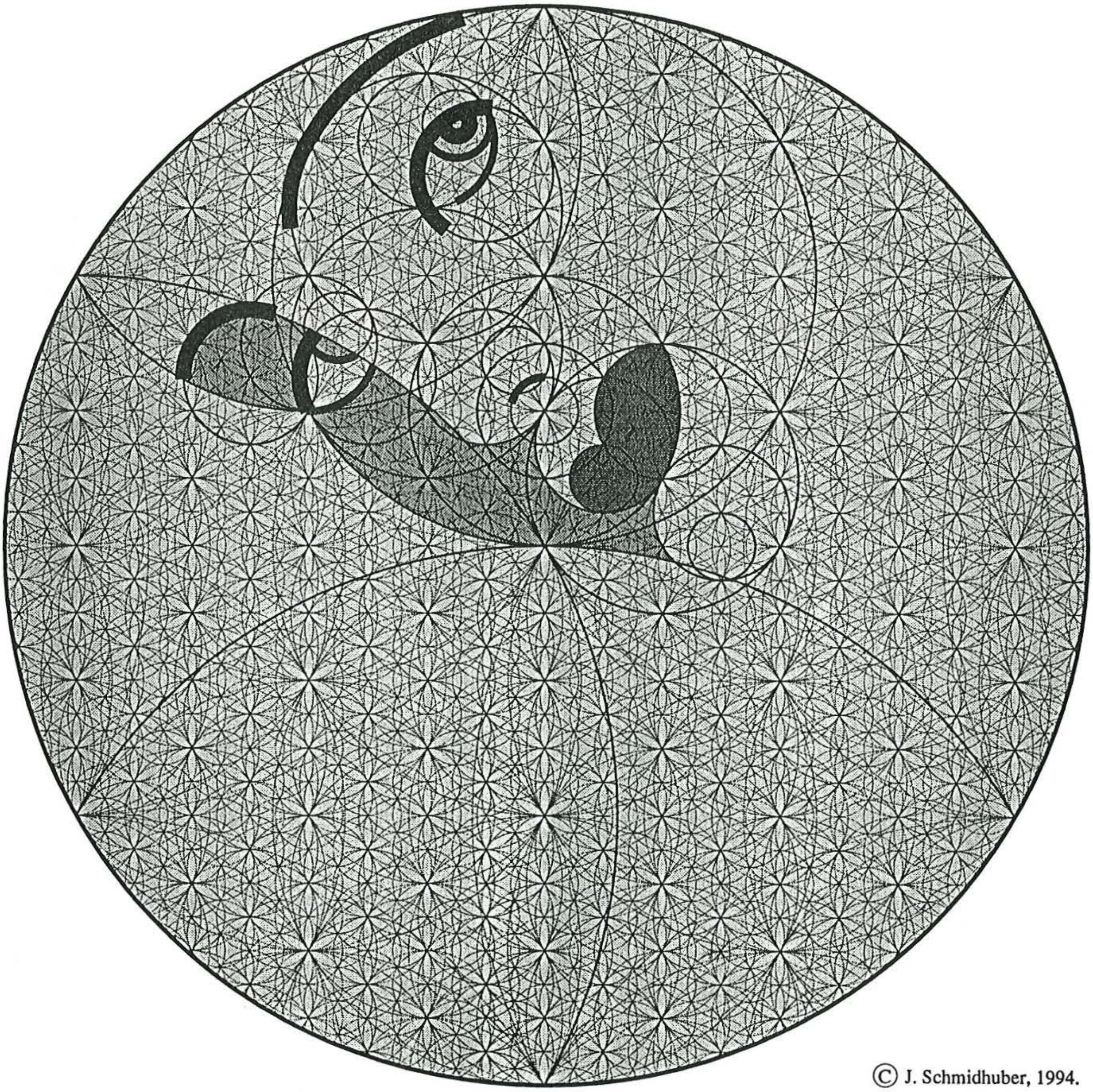

Figure 3: Explanation of the low Kolmogorov complexity (or algorithmic simplicity) of Figure 2. All circles shown are taken from Figure 1. Very few of them, however, are needed to specify the drawing (they are drawn with a thicker line width). Many of them are big and do not require many bits to be specified (compare the coding scheme in section 3.3). *Some comments:* Cheek, forehead, and part of one eyebrow (drawn with a huge legal line width) are shaped from equally-sized legal circles. The bridge of the nose is made from a legal circle twice this size, and so is part of the chin. The basic shapes of the eyes are constructed by two circles each. They happen to fit an ancient rule: "The distance between the eyes equals one eye-width". Many contours of the eyes, nose, mouth, and chin repeat themselves again and again, thus "fitting each other" in a certain sense.

Figure 4: Cartoon of a pin-up girl for a comic book. The face is the one from Figure 2. Again, only Rules I-VI (see section 3) were used to design the cartoon. The drawing has low Kolmogorov complexity and is "algorithmically simple" in the sense that it can be specified by a short sequence of small integers, using the coding scheme from section 3.3. The algorithmic simplicity of the cartoon is explained by Figure 5.

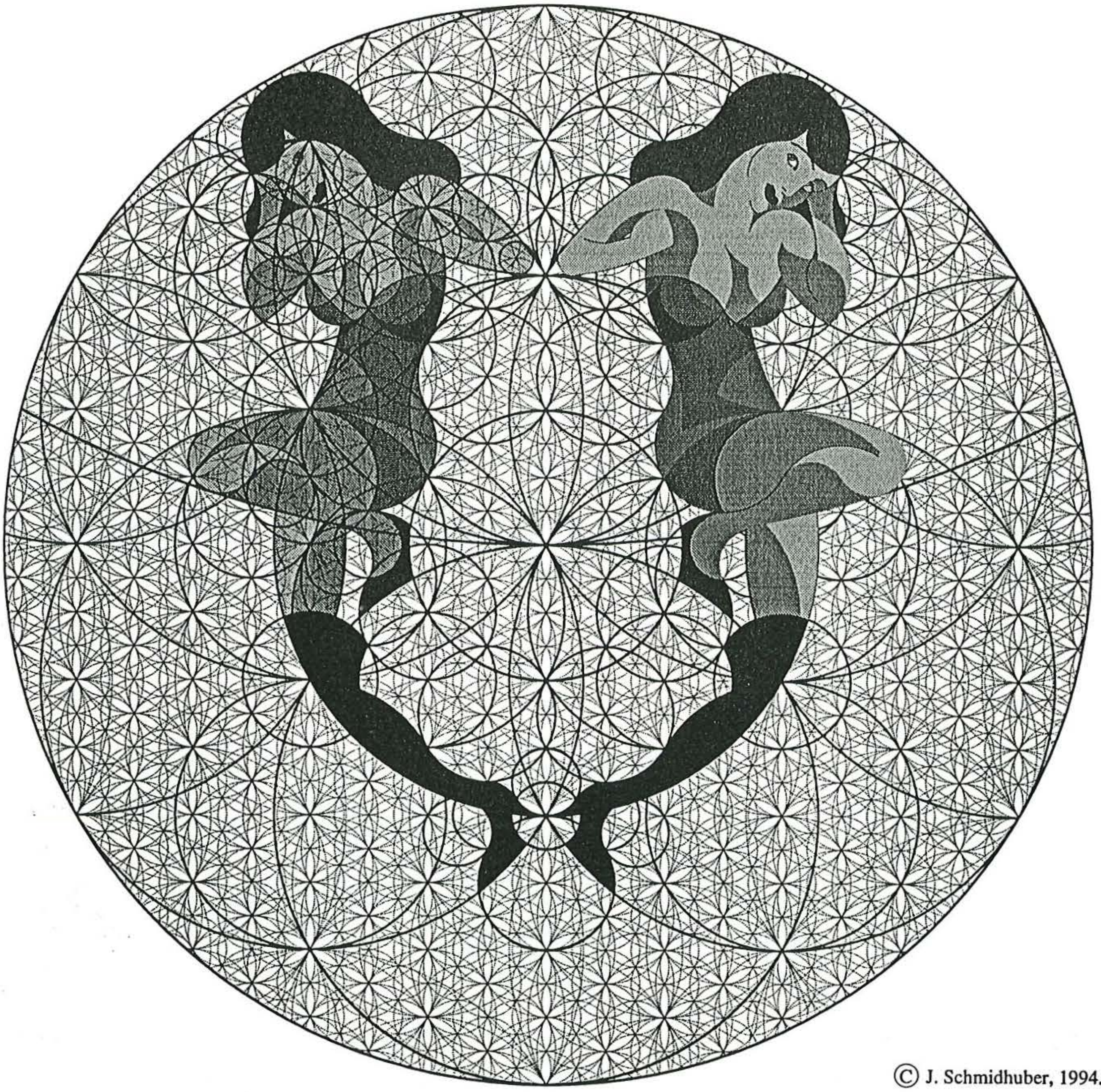

Figure 5: Explanation of the low Kolmogorov complexity (or the algorithmic simplicity) of Figure 4. Most legal circles shown are not necessary to specify the drawing – the dotted ones are shown only to provide a frame of reference (compare Figure 1). Circles actually used are drawn with a thicker line width. Take a little time to study them. *Some comments:* The initial circle is the one defining the shin of the bent leg. Its radius is twice the radius of the circle used for the frame. Note the symmetry of calf, bottom, and waist. They are shaped by a row of three touching, equally-sized legal circles. Shoulders, cheek and forehead, and most parts of the chest, are all defined by circles of the same size, thus creating numerous mirror and rotation symmetries. Observe that the leg with the boot is shaped by "legal waves" of decreasing amplitude. Note that all circles used for the face (see again Figure 3 for details) correspond to legal circles of the more complex Figure 5.

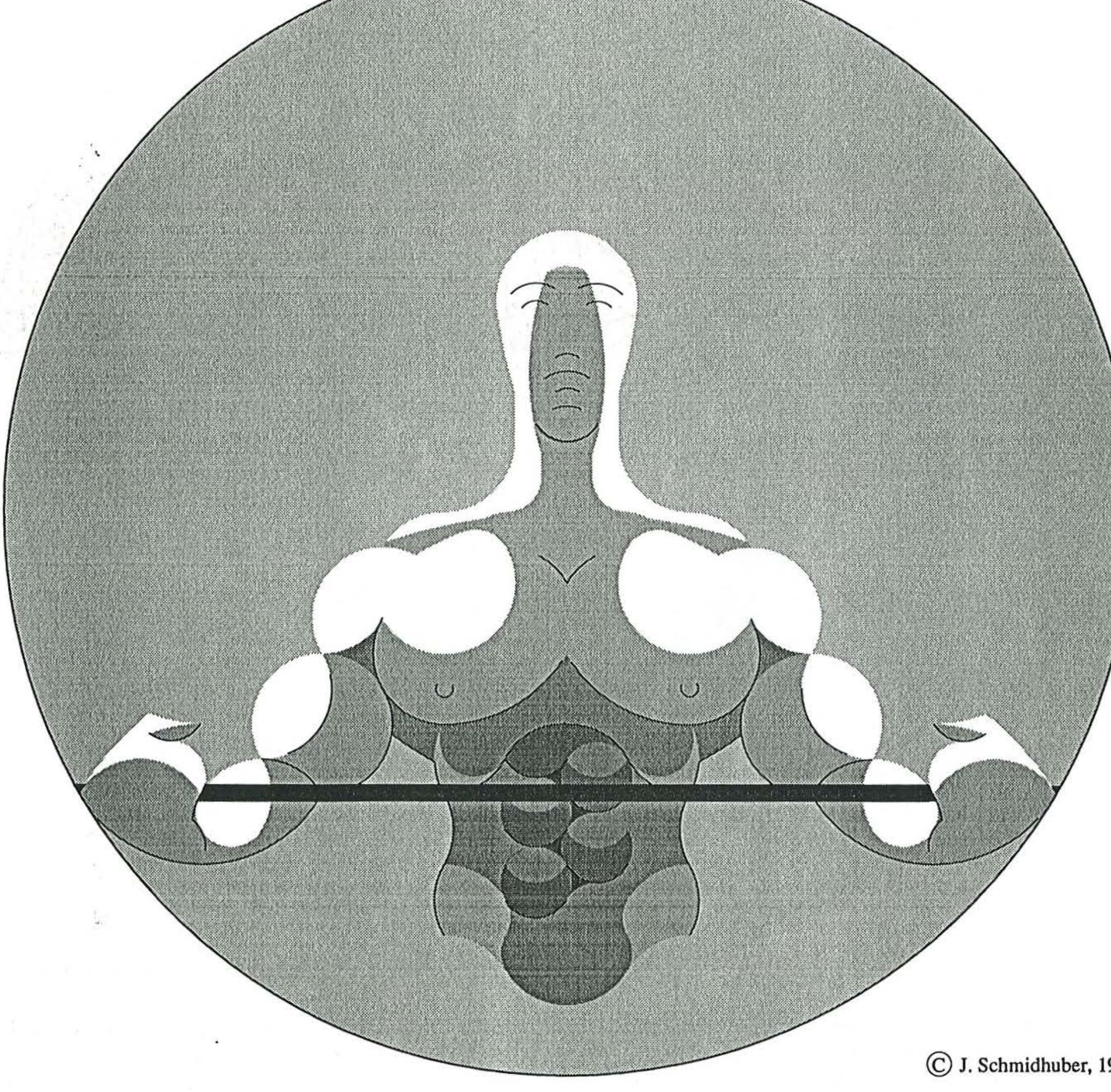

Figure 6: Cartoon of a weight lifter's upper body, designed for the logo of a gym. Again, only Rules I-VI were used to specify the cartoon. Its low Kolmogorov complexity (or its algorithmic simplicity) is explained by Figure 7.

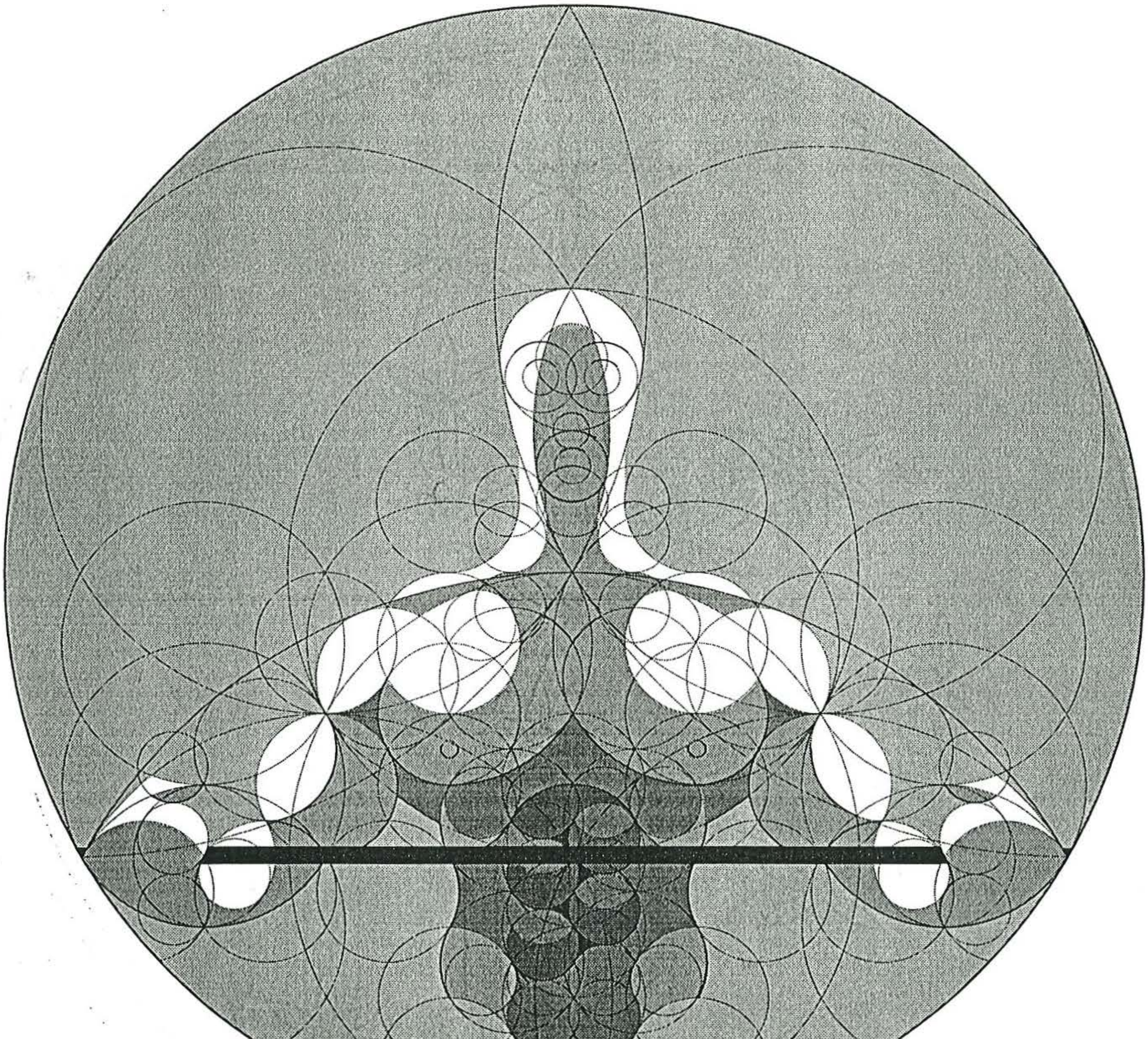

Figure 7: Explanation of the low Kolmogorov complexity (or the algorithmic simplicity) of Figure 6. All circles shown are legal circles taken from Figure 1. Compare the coding scheme from section 3.3. *Some comments:* Note that shoulders and biceps/triceps are shaped by circles of equal sizes. This may be viewed as an idealization of what can be observed with certain human weight lifters. The same circle size is used for many additional features, such as the top of the head, parts of the chest, etc. Mirror symmetry is broken only for the abdominal muscles. The visible part of the dumb-bell belongs to a legal circle with nearly infinite radius (drawn with a huge but legal line width).

Figure 8: 10 flowers with 6 leaves each, makes a total of 60, for J. Sickinger's 60th birthday on April 2, 1994. Again, only Rules I-VI were used to generate the drawing. It can be specified by a short sequence of small integers, using the coding scheme from section 3.3. The algorithmic simplicity (or low Kolmogorov complexity) of the drawing is explained by Figure 9.

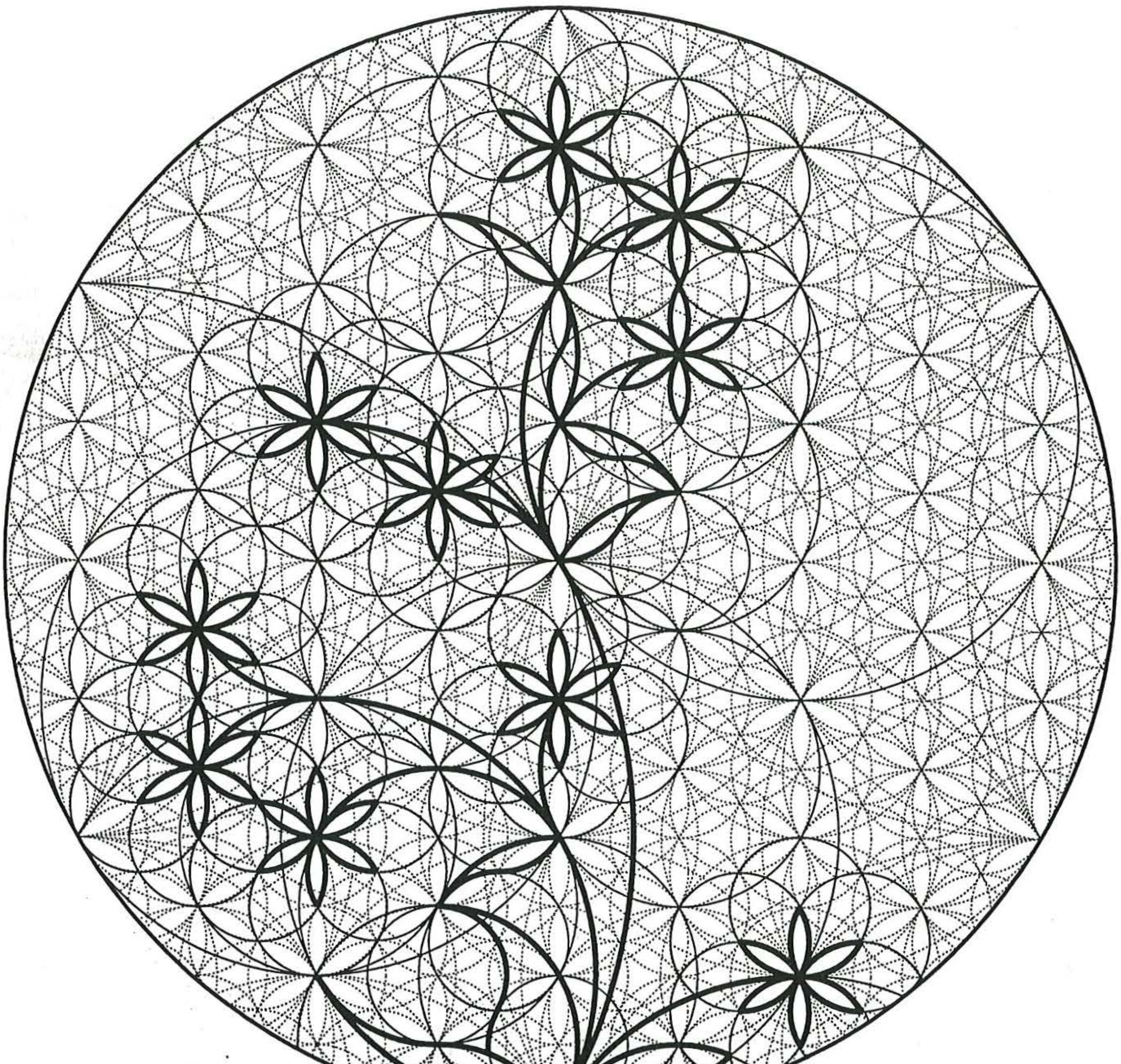

Figure 9: Explanation of the algorithmic simplicity (or low Kolmogorov complexity) of Figure 8. Compare Figure 1 and the coding scheme from section 3.

FKI-197-94 Jürgen Schmidhuber: Algorithmic Art

ISSN 0941-6358