G-pictures: Elements of a Probabilistic Pictorial Grammar

1. *Introduction*

The operation of a computing machine is related to the generation of language by a formal grammar. Language and grammar represent a semiotic system. It was shown that a set recognized by a Turing machine is equivalent to a language generated by a particular type of grammar (phrase-structure, or type 0), i.e., it is recursively enumerable. The formal language approach I am taking here implies semiotic awareness. It renders computational processes of computing machines as sign processes, sometimes referred to as symbolic computation. The state of a computation is, so to say, frozen (synchronic dimension of the sign); and the record of this state is a representation of a picture (sometimes even called *a picture*). Randomness, quite often seen as just a convenient way of generating a well-dispersed set of numbers (instead of doing it by hand), is associated with language generation processes in view of a desired complexity, i.e., qualities through which sentences (in this particular case, images) are considered interesting. From a small set of generating rules, various pseudo-random number sources (actually routines) supply strings of sufficient length and complexity, thus modulating the computed form. The congruential generator that I chose produces a sequence of residues of the modules equal to the word size of the computer. However, since we actually take advantage of finite sequences, the question to be posed is: What is the sufficient level of complexity at which people's aesthetic requirements are met? Primitive interpreters tend to be overwhelmed by quantity. Obviously, my question is not how to produce better random number generators, but rather how to better model aesthetic requirements within the structure of the formal representation. It is along this line that I will first present a probabilistic pictorial grammar, then suggest an improved fuzzy model; finally, also only as a suggestion, a possibilistic approach, i.e., a computational model with an intrinsic semantic dimension.

2. *Generative grammars*

What is a Chomskyan, or a phrase-structure, grammar? It is an attempt to give a formal description to natural languages, assuming that the syntactic aspect of such languages is dominant. As a formal model, a Chomskyan grammar is at the same time a language-generating device, the reason for which the label "generative grammar" was introduced. The four types of grammars—Chomskyan (hierarchy, phrase-structure, class 0); context sensitive; context free; right-linear—can generate four classes of languages. From among these four classes, one class, known as type 2 grammar, i.e., context-free, will be the one that I actually use. It has the advantage of having/being a good approximation to/of the syntax of programming languages and is accepted by pushdown automata.

The Chomskyan grammar (CG) typology can be expanded and new grammars can be introduced. These preserve some properties as defined by Chomsky, although they depart from CG. Examples already acknowledged:

a) *Lindenmayer grammars,* without terminal symbols
b) *Pictorial grammars,* extensions of CG, in which the terminal vocabulary is made up of geometric entities and the concatenation is "regulated" by certain operations of a geometric nature-generation of multi/polydimensional structures (Chang, 1971)
c) *Parametric grammars,* among others.

2.1. Object of the study

The object of my study is already a classic for computer scientists dealing with the issue of describing and generating images: Mondrian's aesthetic system (known under the name *Suprematism)* as expressed in a precise segment of his creative activity through his paintings, and the theory he developed. Mondrian's geometric construction in orderly fields—non-monotonous, non-serial—is quite stimulating. It is part of the general development through which after all computers were designed and built. General principles of the aesthetics he defined are:

a) The visual means of expression should be the plane or rectangular prism of a primary color (red, yellow, blue) and non-color (white, black, gray).
b) Composition itself becomes aesthetic expression.
c) Color is opposed to non-color (white, black, gray).
d) An equivalence of the means of expression is necessary.

Of different dimensions and of different colors, the expressive components should have the same value. In general, balance is based on a bigger surface of non-color or empty space and a small surface of color or matter.

Due to the strong formal nature of Mondrian's work, scholars found in it ready examples for descriptions based on information processing. A general aesthetics of his work could also be easily derived and defined. Explaining or describing Mondrian does not necessarily explain or even describe art in general. But some clues are offered regarding what it takes to describe images and how complex the issue of interpretation is, even if we involve in our approach such a powerful information-processing device as the computer.

Previous approaches include:
1. Graph theory
2. Statistical analysis: Mondrian's *Key and Ocean* (1915) inspired such analysis for description and generation of replicas. Noll's study (1966) revealed that a "new Mondrian" was preferred by 55% of the audience. Noll considered the complexity of the proportions of Mondrian's paintings; the components and their relations; the probability of color, among other criteria. "They [the audience] strongly associated randomness with human creativity and therefore incorrectly identified the Mondrian as the computer generated picture."
3. Web grammars (cf. Pfalz, Rosenfeld)

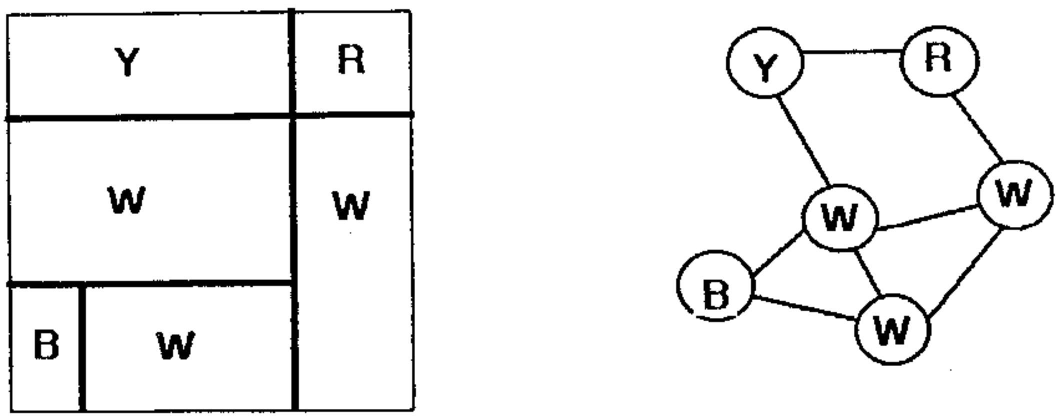


Fig. 1

4. Metric-statistical (cf. G. Fischer, University of British Columbia)
5. Structural-topological (cf. Anthony Hill).

2.2. Pictorial grammars

In my approach, I will make use of a still new class of generative grammars—pictorial grammar—along the line defined by Shi-Kuo Chang (1973), complemented at the beginning by a probabilistic component, and later by a possibilistic type of description. This approach makes use of the strong formalism developed around the paradigm of language and allows for future developments in which symbolic computation plays the main role in describing and interpreting complex visual configurations.

Let us consider the following grammar:

$$G = (V_N, V_T, S, R)$$

in which

V_N = non-terminal vocabulary (set of non-terminal symbols)
V_T = vocabulary (set of terminal symbols)
$V_N \cap V_T = \varnothing$
$S \in V_N$ initial symbol of the grammar G
$R = (R_1,...R_N)$ set of production ("re-writing") rules, of the form $\alpha \rightarrow \beta$, in which R_i, with i = 1 has the form V $\rightarrow R_I (X_1,...X_{ni}), X_i (V_N \cup V_T), V \in V_N$

Each rule R_i is represented by a picture.
Each picture is subdivided in n_i regions.
Each symbol X_i, (in which i = 1). is attached to a region (X_1 = region 1, X_2 = region 2, etc.)

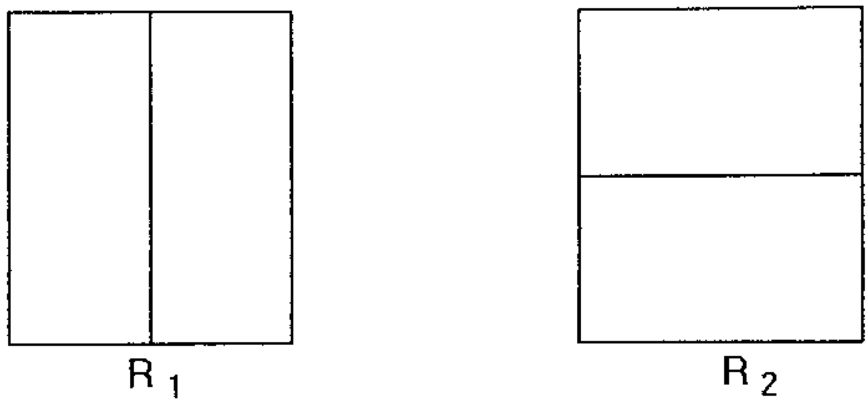


Fig. 2

Another case, with A, B, C, D as terminal symbols

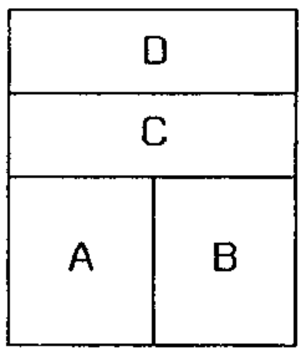


Fig. 3

can be represented as follows:

$S \Rightarrow R_2(V, W) \Rightarrow R_2(R_2\,(D, C)W \Rightarrow R_2(R_2\,(D, C), R_1\,(A,B))$

Evidently: $A, B, C, D, \in V_T; V, W, \in V_N$

or: $Y : R_2(R_2\,(D, C), R_1\,(A,B))$

The terminal vocabulary can consist of colors (of the region) or other qualitative characteristics. If we use only one terminal symbol, we can get a sectoring of the field:

$Y : = R_2(*, *), R_1(*, *))$, in which $* \in V_T$, and Y- - is the structure

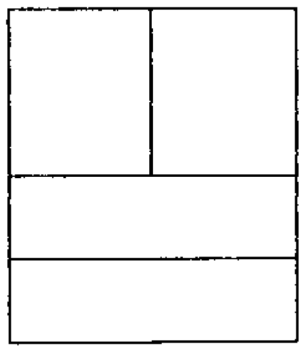


Fig. 4

Using other rules, we can decompose the non-terminal symbols/vocabulary in as many regions we want, bordering any kind of mathematically describable curve. Example:

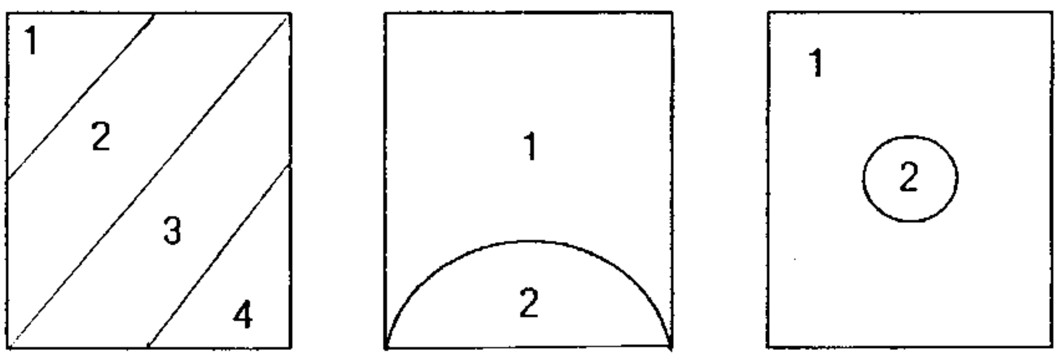


Fig. 5

If G is a pictorial grammar thus defined, let L(G) be a language generated by G. We shall call *G-painting* each element of L(G).

In order to generate G-paintings on computers, let us first define *probabilistic pictorial grammars.* In a probabilistic pictorial grammar, each rule R_i is given together with a stochastic vector whose jth-component indicates the probability that the rule R_j is applied after R_i. Also provided is a probabilistic distribution over the set of rules (cf. Salomaa, 1969). For a pictorial grammar, we should consider an initial probabilistic distribution and for each non-terminal symbol of each rule, we will have a stochastic vector.

The language generated by a probabilistic grammar consists of all the phrases P generated by the grammar, so that the probability associated to the generation of P is greater than a threshold defined at the beginning. If the threshold is greater than 0, L(G) is a finite language. We shall define the threshold = 0, but shall limit the number of decompositions used for the generation of a G-picture. Hence, the generation process (generative process) and the language are *finite*.

Using a pseudo-random multiplicative-congruent generator to obtain, through simulation, the words of a language L(G), we shall implement a G-probabilistically determined pictorial grammar.

Properties:

1. These grammars are context free (or type 2, in Chomskyan typology); i.e., they are closed under union.
2. They can be ambiguous or not.
3. Each grammar of type 2 is equivalent to a grammar of the same type in which each rule containing a terminal symbol has the form

$$V \to A,\ V \in V_N,\ A \in V_T$$

4. Property 1 ensures that we can define unions of such grammars and that we can decompose them into simpler grammars.
5. It can be *decided* (decidability) whether a given chromatic or achromatic field is a G-picture (i.e., can be generated from the grammar G).

3. *Possibilities for application*

3.1. Analytic approach (to analyze chromatic and nonchromatic fields)
3.2. Storage in a computer memory of the words accepted (G-pictures)

Ex: The structure Y

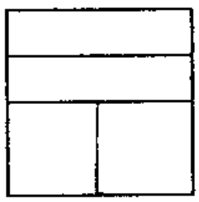


Fig. 6

is defined as

$$R_2(R_2\ (D,C),\ R_1(A,B))$$

or more generally as

$$R_2(R_2(*,\ *),\ R_1(*,\ *))$$

Once stored, such descriptions can be manipulated according to image-processing, image-generation, or even image-interpretation needs:

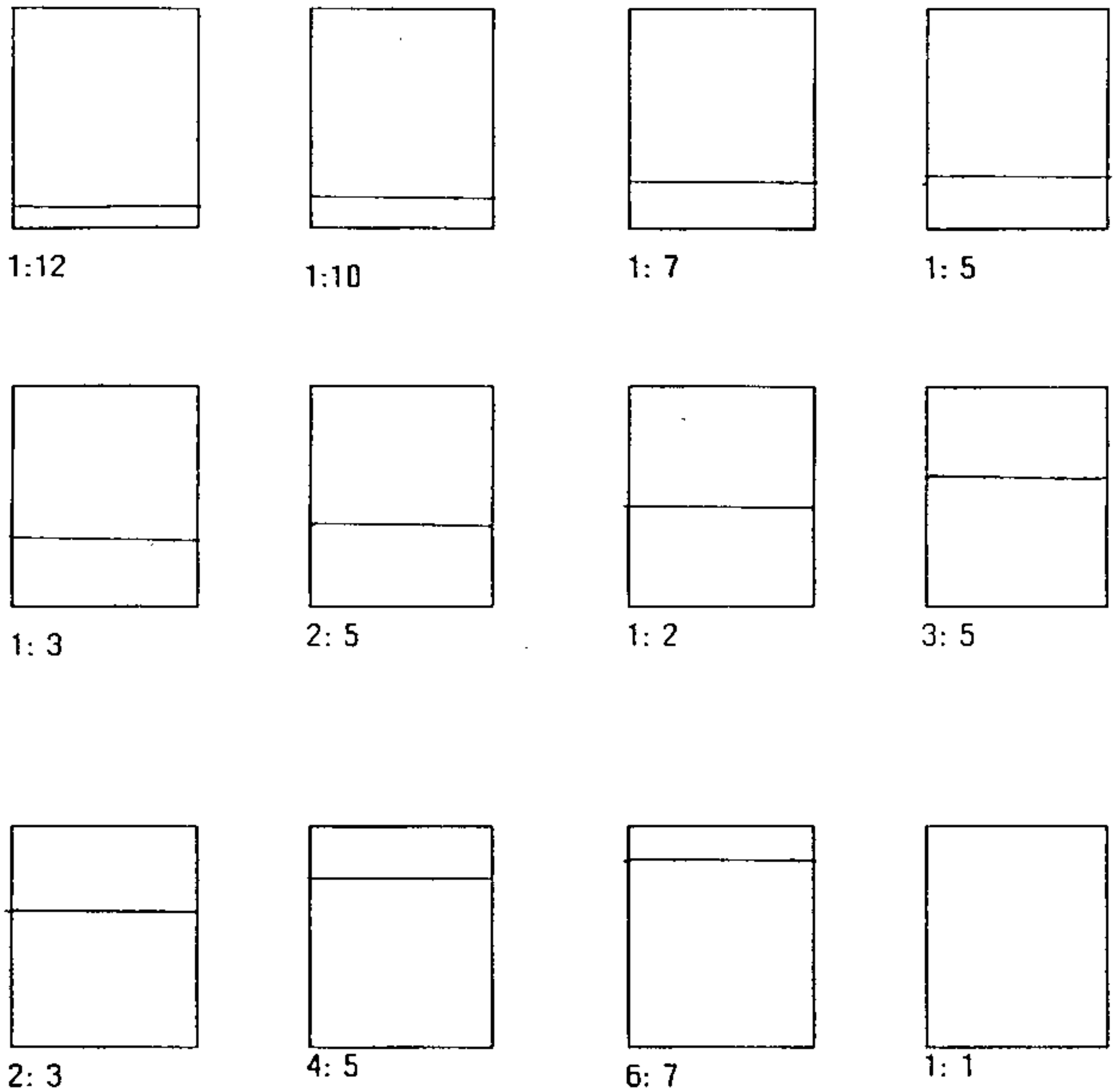
3.3 Optimal cut of metal sheets, patterns
3.4 Pattern recognition
3.5 Simulation of works of art (*re-writing*)
3.6 Introduction of *control words* and derivations: We can generate $L_A(G)$-language controlled by the words A = the set of control words (restrictive specifications). The words in A can be aesthetically relevant or can express some other goals.
3.7 Calculation of the probability of each G-picture for a given grammar G.

Mondrian Proportions:

1:12, 1:1O, 1:7, 1:5,

1:3, 2:5, 1:2, 3:5,

2:3, 4:5, 6:7, 1:1



1. The rules of sectioning (horizontal/vertical) expressed by these proportions will contribute to the definition of the probabilistic pictorial grammar characteristic of Mondrian.

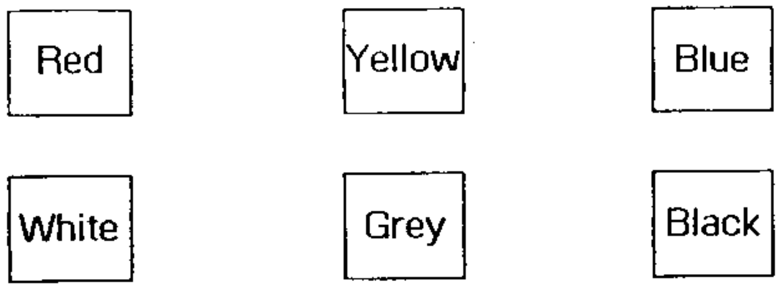
2. Rules of the form

$$V \rightarrow A,\ V \in V_N,\ A \in V_T$$

in which

$$V_T = \{\text{Red, Blue, Yellow, White, Grey, Black}\}$$

(Mondrian used only pure colors and achromatics.)



Hence:

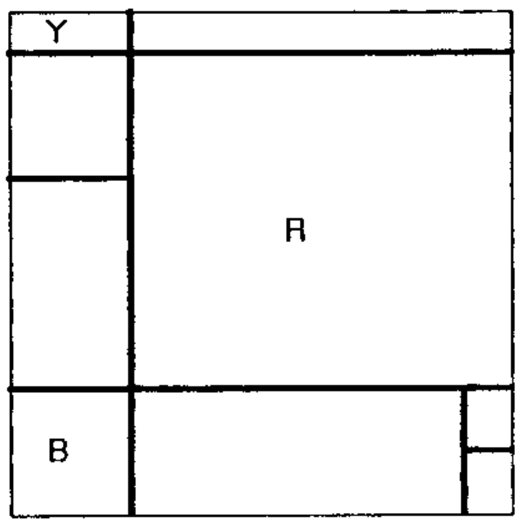
$G = (V_T, V_N = \{S\}, S, R)$, the rules being
$S \rightarrow R_i\ (S, S)$; i = 1,24 (24 corresponds to the 12 proportions)
$S \rightarrow A,\ A \in V_T$

In other words, we have 30 rules.

The grammar is attached to an initial probabilistic distribution, i.e., a stochastic vector of 30 elements and 24 x 2 stochastic vectors, each of 30 elements.

Generating G-paintings on the computer, we can obtain simulated Mondrian images:

Red, Blue, Yellow—specified by λ (wave length)
White - Grey – Black—three grades of saturation in logarithmic progression



The next problem is whether the stochastic pictorial language thus generated is closed under generalized sequential machines (GSM), which can be understood as generic compilers. Since it is known that the mappings defined by such machines generally preserve the typology of Chomsky's hierarchy of languages, it makes sense to assume that successive translations of L by GSM mappings will lead to a language of the same type as L. Mondrian's hope was that his aesthetics prove context independent. At least in terms of formal languages, this seems to be the case.

4. *Future directions*

A. Use of fuzzy pictorial grammars
B. Transition from a probabilistic to a possibilistic level, i.e., semantic implications

4.1 A fuzzy grammar is also defined by a quadruple:

$$G = (V_N, V_T, S, R)$$

in which the definitions of the elements are basically the same as defined in the introduction.

V_N is a set of labels of certain fuzzy sets on V_T * called fuzzy syntactic categories. The elements of R are expressions of the form

$$\alpha \overset{S}{\rightarrow} \beta, \quad S \in [0,1]$$

in which

α and β are strings in $(V_T \cup V_N)^*$
S is the degree of membership of β given α
A fuzzy grammar G generates a fuzzy language L (G):
A string x of V_T^* is said to be in L (G) iff x is derivable from S.
The degree of membership $\mu_G(x)$ of x in L (G) is

$$\mu_G(x) = \text{supmin } (\mu(S \rightarrow \alpha_1), (\mu(\alpha_1 \rightarrow \alpha_2), \ldots (\mu(\alpha_m \rightarrow x)) > 0$$

in which

$(\alpha_i \rightarrow\text{-}\alpha_{i+1})$ is the non-null S_{i+1} such that

$(\alpha_i \overset{S_{I+1}}{\rightarrow} \alpha_{i+1}) \in R \ \forall_i = 0, m$
if $\alpha_0 = S$ and $\alpha_{m+1} = x$

Two considerations follow:

4.1.1 Using a fuzzy language we can evaluate which picture from the generated series is more Mondrian than others (a quality that will be called appropriateness).
4.1.2 We can apply the Suprematist principles to see whether his followers were faithful or only used Mondrian's work as a departure point to develop their own concept (e.g., the Russian Suprematists, especially Malevitch, are a case in point).

All these, as well as other issues, are still to be approached, but the framework is well defined.

4.2 Visual language

At the AI International Conference in Cambridge, 1977 L.A. Zadeh introduced PRUF—A Language for the Representation of Meaning in Natural Languages. This language is intended to serve as a target language for the representation of meaning of expressions in a natural language. I would like to extend it to visual languages. It is based on the premise that the uncertainty in the interpretation of a proposition in a natural language is possibilistic rather than probabilistic in nature.

This implies that a proposition p of the form "X is F", in which X is the name of an object and F is a fuzzy subset of an universe of discourse U, in PRUF translates into an expression P which defines a procedure whose domain is the set of possibly fuzzy relations in a database. Acting on those relations, P yields a possibility distribution which characterizes the information I(p) conveyed by p.

If we take p to be a picture generated by a pictorial grammar and the components (horizontals, verticals, colors, proportions, etc.) *fuzzily* ascertained (as in Mondrian's theoretic concept; for example: balance is based on a bigger surface of non-color or empty space and a small surface of color, with a big and a small, evidently fuzzy by their nature), it follows that a pictorial proposition induces a possibility distribution

$$\Pi_p = L$$

A certain value (proportion) might possibly be achieved, i.e., the possibility of a certain picture is

$$\text{Poss}\,\{p=L\} = \Pi_p\,(L) = \mu_L\,(L)$$

Such characteristics as *near, far,* etc. are translated into PRUF as

{color (red, small);
{
{ $\Pi_{location} = \text{surface}_1^{\text{NEAR}[\text{surface=white}]}$

in which NEAR is a fuzzy relation which associates each pair of surfaces through the degree to which they are close to each other. This language evidently has greater expressive power than that of semantic networks or of first order predicative calculus. It allows us to deal with the imprecision of the visual perception in a systematic manner. The logic underlying PRUF is a fuzzy logic with its base logic originating in Lukasiewicz's LAleph_1 logic in which the truth-values are points in the unit interval. Thus continuity, essential in modeling images, is regained. The use of linguistic quantifiers (*many, few, several, almost all,* etc.) is also important since we describe images in language, after all.

Some years ago (1977), I proved that Peirce's sign definition is a Turing computable function and that fuzziness is intrinsic to it. Obviously, my interest goes along this line, i.e., sign processing as a more general case of symbol processing. Today I reported on an application that requires formalism considered already obsolete. I predict that the computer science community will finally acquire semiotic awareness. This will free us to rediscover and improve other procedures that were abandoned because we actually did not know what to do with them.

References

Chang, S.K. (1971). Picture Processing Grammar and its Applications, in *Information Sciences,* 3, pp. 121-148.

— (1973) An Interactive System for Chinese Character Generation and Retrieval, in *IEEE Transactions on Systems, Man and Cybernetics,* 3

Nadin, Mihai (1977). Sign and fuzzy automata, in *Semiosis,* Heft 1, 5.

Noll, Michael A. (1966). Human or machine-a subjective comparison of Piet Mondrian's 'Composition with lines' and a computer generated picture, in *The Psychology Record,* 16 (1966), pp. 1-10.

Salomaa, A. (1969). Probabilistic and Weighted Grammars, in *Information and Control,* 15.