SYNTACTIC ANALYSIS OF TWO-DIMENSIONAL VISUAL SIGNALS IN THE PRESENCE OF NOISE

M. I. Shlezinger

UDC 007:621.391.19

A formal construction is described for prescribing sets of two-dimensional images, called two-dimensional grammar. A method of two-dimensional programming is proposed for the maximization of an additive similarity function on the set of images, generated by the given grammar.

THE PRESENT STATE OF THE PATTERN RECOGNITION PROBLEM

It is worthy of note that although pattern recognition arose because of the need to solve problems of visual analysis, the majority of publications in pattern recognition at the present time has almost no relation to these problems. At the origin of such widely known methods of recognition as the method of the generalized portrait [1, 2], the potential functions method [3, 4], and the algorithms of Bongard [5, 6] and Rosenblatt [7], it was assumed that they would be used primarily for the solution of visual problems. At the present time these algorithms are successfully applied to the solution of problems of medical diagnosis [8], to predict instrument lifetime [9], and the distinction of oil-bearing and water-bearing strata [10]. At the same time, it is fairly evident that the recognition of visual signals is found beyond the limits of this field. In the opinion of the author, such a situation arose because in pattern recognition for a long time difficulties of one kind were overestimated, and difficulties of another kind were almost completely ignored.

The difficulties of the first kind arose when the set of signals to be recognized, which correspond to given patterns, are not known in advance. To overcome this kind of difficulty many approaches and procedures have been proposed, constituting the theory of recognition learning.

The difficulties of the second kind occur even when the set of signals to be recognized is completely and uniquely given, but an algorithm is not known that for an arbitrary signal would indicate if it enters into a given set or not.

The very widespread opinion that if the set is given in some way, then the problem of determining membership of some element in this set is trivial, is deeply erroneous. This opinion is a result of misapprehending the fundamental difference between unique prescription of a set and an algorithm defining membership in this set. The essential difference between these concepts is at least confirmed by the fact that classical algorithmically undecidable problems are formulated precisely as problems of recognizing membership of an arbitrary element of a completely defined, uniquely given set of elements [11]. Another example of nontriviality of the passage from set to algorithm can be the situation where a certain set is defined as a set of functions, whose maximum exceeds a given threshold. Although for many subclasses of functions this problem is not algorithmically undecidable, it is, however, obvious that the problem of finding an algorithm, testing for an arbitrary function its membership in this set, is far from trivial.

The specific characteristic of the problem of recognizing visual signals consists precisely in that in the majority of cases the set of signals belonging to a single class is known in advance and can easily be given by means of some constructive procedure, for example, by means of the method of feasible transformations [12, 13]. In this case the recognition training problem does not even arise, and the problem of recognition proper comes to the fore, i.e., determining the membership of a given signal in a certain predefined set.

To a still greater degree, the specific characteristic of problems of visual analysis, their exclusivity among other recognition problems, finds expression in the approach called "linguistic" [14]. The algorithms of this approach are intended basically for the recognition of visual images, and in this is their great advantage over algorithms that are as suitable for prediction of the causes of forest fires as for the classification of fossil insects [15]. However, all the work in this direction, as considered by Martelli and Montanari [16], has still not formed a theory but, rather, a "repertoire of little tricks," heterogeneous in their methodology and aims. This evaluation, given in [16], is excessively severe, but examples confirming it are much more numerous than those contradicting it. Contour enhancement, noise smoothing, determination of image skeletons, finding the edges of images, normalization of

Translated from Kibernetika, No. 4, pp. 113-130, July-August, 1976. Original article submitted April 28, 1975.

geometrical dimensions – this is a by-far-from-complete list of the "little tricks" that to one degree or another are encountered in any practical recognition algorithm [17]. On the other hand, recent theoretical work in the linguistic direction has an explicitly expressed tendency to investigate ever more general models and, naturally, does not become more comprehensive to the developers of practical recognition systems.

The rapid development of the theory of the linguistic approach, in the absence of a sufficiently useful list of solutions of practical problems, conceals the danger of a gap between theory and practice of recognition – the danger that just those difficulties that merit, possibly, the most sustained attention, will pass unnoticed. It is possible to state three of the most significant defects characteristic of certain work in the linguistic approach to recognition. First, the algorithms describing the images do not directly analyze the visual signal, but rather some verbal description of it, which in itself is no longer the image, but the result of a far-from-trivial transformation of the image. Secondly, the constructions proposed in certain work to generate the set of images do not take into account the probabilistic character of visual images. The third defect consists in the one-dimensional character of well-known generative grammars, ill-adapted to the description of classes of two-dimensional images. Of course, not all the known work has all of these defects, but there is no work that does not have at least one of them. Let us consider each of these defects in greater detail.

1. In order that some algorithm for the syntactic analysis of images be able to be considered to be applicable in practice, it is necessary that the values of the symbols which compose the analyzed word be able to be obtained by simple physical measurements, effected on the image itself. This simple and fairly obvious requirement is not satisfied by many publications of the linguistic direction. For example, in [18, 19] it is assumed that the image to be recognized has already been represented in the form of a word, giving information about the structural elements (rectilinear segments, arcs, etc.) composing the image and their relative positions. These publications are useful and interesting in many respects, but the algorithms proposed in them do not effect visual analysis. An algorithm of visual analysis should have at the input precisely the visual signal or the result of its simple transformation, such as, for example, the measurement of the mean brightness of individual portions of the image. In any case, the description of an image in terms of segments, arcs, and their relative positions, cannot be considered to be the result of a simple transformation of the image.

2. Formal grammars, which are a method for prescribing certain sets, and, specifically, sets of words of some given language, are not completely adequate to the corresponding visual concepts. As a rule, the assignment of some image to a given pattern cannot be effected with the same degree of definiteness as is possible for the assignment of some word to a given language. The concept of "pattern" in the recognition of visual images should be to the same degree a generalization of formal language as the concept of fuzzy set [20] is a generalization of the simple concept of set. In this respect a very fruitful approach is that of Kovalevskii [21], who considered a random mechanism acting on an observed image after it has been obtained by formal grammatical rules. It can also be useful to introduce into the language probabilistic structure, as is done by Fu [22]. The work of Kovalevskii and Fu substantially reduces the gap between theory and practice of recognition and leads directly to certain practical results (e.g., [23, 24]). However, the stochastic grammatical constructions proposed by them have the third of the above-listed defects.

3. Classical formal grammars, even if they are completed with some kind of probabilistic structure, cannot be considered to be adequate to the overwhelming majority of problems of visual analysis because of the "one-dimensionality" of these grammars. The concept of "visual pattern" must be given by some kind of construction, similar to a grammatical one, but the words generated by this construction must be two-dimensional arrays of symbols, and not one-dimensional sequences, as occurs in ordinary languages. Of course, at first glance it may appear to be indifferent to in what form the image is given – in the form of one- or two-dimensional arrays. Of course, any image, by the application of a rectangular or any other scan, can be represented in the form of a one-dimensional sequence. However, in one-dimensional representation of an image the relationships of geometrical closeness between individual elements change, and such important relationships for visual analysis as symmetry, connectivity, etc., are lost. Just those properties are lost that make an image a single whole, and not just a simple set of abstract variables.

At the present time fairly many attempts are known (basically, by non-Soviet authors) to obtain constructions similar to formal grammatical ones, but which take into account the two-dimensional character of the generated words. Expressive names (mosaic grammars [25], matricial [26], parallel [27], web [28]) at least partially characterize their specific character. Unfortunately, these grammars are more adapted to the generation of images than their recognition. This means that the algorithms verifying membership of an image in some two-dimensional language are very complicated and the absence of such algorithms has even been demonstrated for certain classes of grammars. At the same time these grammars have to the full degree the second of the listed defects, i.e., they completely neglect the probabilistic character of the generated images. Nevertheless, the appearance of two-dimensional languages, in response to the needs of visual analysis, allow it to be hoped that such two-dimensional languages will be an acceptable formalism for the basic concepts of visual analysis. The theory of two-dimensional languages can become the mathematical foundation of image recognition and thereby overcome the gap presently filled by various kinds of "little tricks." The development of the theory of two-dimensional languages must be considered promising also because this theory can point the way to the construction of new computers that will be characterized by an intrinsically two-dimensional and parallel organization of the computational process, qualitatively different from that adopted in current computers.

The present article is an attempt to combine the positive aspects of two groups of investigations. The first group contains investigations of various two-dimensional generalizations of formal grammars. The second group is composed of the cited work of Kovalevskii, in which methods are proposed for the syntactic analysis of images generated by ordinary, i.e., one-dimensional, grammars, but observed in the presence of substantial noise influence.

Following Kovalevskii's methodology [29], in the present work by means of a construction G, similar to a formal grammatical construction, we will prescribe not the entire set X* of images, belonging to some visual pattern, but only some negligible part of it, called the set V*(G) of ideal, or reference, patterns. To any ideal image $v \in V^*(G)$ there will correspond a certain set of real images, similar to v. This set is fuzzy in the sense of Zadeh [20], and therefore the membership function $f_v(x), x \in X$, to this set, called a property, takes on values not necessarily equal to 0 and 1.

The concept of "pattern" is defined as a fuzzy set of real images, close to at least one of the ideal images. This means that the pattern is defined as a fuzzy set with membership function $f_G(x) = \max\limits_{v \in V^*(G)} f_v(x)$.

The recognition problem is formulated as the problem of calculating the value of the function $f_G(x)$ for the analyzed image.

The problem of syntactic analysis of an image x (in the first approximation) consists in finding an ideal image, generated by a given construction G, and maximizing the similarity function, i.e., in finding an image $v(x) = \arg \max\limits_{v \in V^*(G)} f_v(x)$. This problem is formulated in more exact terms below.

The problem of image classification arises when several grammars $G_1, G_2, \ldots, G_n$ are given, and consists in calculating the value $K(x) = \arg \max\limits_{i} \max\limits_{v \in V^*(G_i)} f_v(x)$.

The construction proposed in the present article for prescribing the set of ideal images will be called a two-dimensional grammar.

The method that is proposed for calculating the membership function $f_G(x)$ will be called a two-dimensional program.

TWO-DIMENSIONAL GRAMMARS

We shall consider two alphabets to be given. The first is called the signal alphabet and is denoted by the letter V. From its elements are directly composed the images that are defined as the function v, given on the finite set Tv – the visual field – and taking on values in the set V. In this sense the signal alphabet is fundamental. It is in some degree analogous to the alphabet of terminal symbols in one-dimensional grammars.

In addition to the signal alphabet, we shall consider an alphabet S of structural elements. The structural elements do not enter into the image, they serve for the construction of feasible images. In this sense the alphabet of structural elements is auxiliary, analogous to an alphabet of nonterminal symbols. From the structural elements is directly composed the description of an image, defined as function s, given on a finite set Ts – the description field – and taking on values in the set S.

In simple cases the description field is isomorphic to the field of vision. However, in the general case this is not necessarily true.

An element of the alphabet V will be denoted by v, and an element of the alphabet S by s. We shall call the union of the visual field and the description field a field, and denote it by T, and an element of the field, a cell, and denote it by the letter t. By Z we shall denote the set $V \bigcup S$, whose element is called a symbol and denoted by z. We shall term by variant z a pair (v, s) "image – description." This means that a variant is a function, given on the set $T = Tv \bigcup Ts$ and taking on values in the set Z, and in such a way that $z(t) \in V$ if $t \in Tv$ and $z(t) \in S$ if $t \in Ts$.

Most important for two-dimensional grammars is the concept of feasible variant. To formalize this concept we shall introduce some further definitions.

We shall consider two cells $t \in T$ and $t' \in T$ to be adjacent if a certain predicate R(t, t'), fixed for the given grammar, is equal to 1. In the present article the predicate R will everywhere be assumed to be symmetrical. Furthermore, it is everywhere assumed that a cell is not adjacent to itself, i.e., $R(t, t') = 0$ for all $t \in T$.

We shall denote the set of pairs of adjacent cells by N. This means that the notation $(t, t') \in N$ is equivalent to the notation $R(t, t') = 1$.

For every pair of adjacent cells t and t' we shall assume given the sets $ZZ(t, t')$ of feasible pairs (z, z') of symbols $z, z' \in Z$. We shall call a variant z feasible if for every pair $(t, t') \in N$ the relation $(z(t), z(t')) \in ZZ(t, t')$ holds.

An image v^* is called feasible if there exists a feasible variant $z = (v^*, s)$. If the variant (v^*, s) is feasible, then the description s is called a possible description of the image v^*.

A two-dimensional grammar G is an ensemble of these objects, i.e., the sextuplet $\langle V, S, Tv, Ts, R, \{ZZ(t, t'); (t, t') \in N\}\rangle$.

For the subsequent exposition we require sets $Z(t)$, $t \in T$, each of which denotes a set of symbols that is feasible in the cell t. As a rule, this set contains all those z for which $(z, z') \in ZZ(t, t')$, but other situations can also apply. The sets $Z(t)$ and $Z(t')$ can intersect and coincide.

We introduce the notation σ for the pair (z, t) and two functions $t(\sigma)$ and $z(\sigma)$, where $t(\sigma) = t((z, t)) = t$, $z(\sigma) = z((z, t)) = z$. For every $t \in T$ we denote by $\bar{\sigma}(t)$ the set of pairs of the form (z, t), where $z \in Z(t)$. We denote by $\bar{\sigma}$ the set $\bigcup_{t \in T} \bar{\sigma}(t)$. It is understood that the sets $\bar{\sigma}(t)$, $t \in T$, thus constructed have the following property: for arbitrary $t' \neq t''$ the sets $\bar{\sigma}(t')$, and $\bar{\sigma}(t'')$ are disjoint. By analogy with the set $ZZ(t, t')$ for adjacent cells t, t' we bring into consideration the set $\overline{\sigma\sigma}(t, t')$, consisting of those pairs (σ, σ') of the form $((z, t), (z', t'))$ for which $(z, z') \in ZZ(t, t')$.

An element $\sigma \in \bar{\sigma}(t)$ will be called a node lying in the cell t, an element $(\sigma, \sigma') \in \overline{\sigma\sigma}(t, t')$ an arc, joining the nodes σ and σ'. If a certain pair (σ, σ') is such that $R(t(\sigma), t(\sigma')) = 1$, does not belong to the set $\overline{\sigma\sigma}(t, t')$, we shall say that this pair is forbidden or that the nodes σ and σ' are not joined by an arc.

LOCALLY CONJUNCTIVE PREDICATES

The feasibility of a variant is in fact a locally conjunctive predicate of bounded order on the set of variants [30]. This means that in the set T are distinguished certain subsets $T_i \subset T$ consisting of a small number of cells. Let $t_{1i}, t_{2i}, \ldots, t_{ki}$ be cells occuring in the set T_i. To each set T_i is associated a predicate m_i, dependent only on the values of the elements $z(t)$, $t \in T_i$.

The term locally conjunctive predicate designates a predicate represented in the form $\bigwedge_i m_i(z(t_{1i}), z(t_{2i}), \ldots, z(t_{ki}))$. The quantity k is called the order of the predicate.

In our case feasibility of a variant is a second-order locally conjunctive predicate on the set of variants. At the same time, the important circumstance that should be kept in mind is that two-dimensional grammars are given by a locally conjunctive predicate on the set of variants, while Minsky and Papert [30] consider locally conjunctive predicates on the set of images. The difference between two-dimensional grammars and the predicates of Minsky and Papert is that their local predicates indicate the feasibility of certain groups of symbols directly composing the image, i.e., groups of signals. In two-dimensional grammars the constraints are imposed on the groups of variables, among which are not only signals, but also structural elements. The latter do not enter directly into the image, but enter only into its feasible description. In relation to the analysis of perceptrons, Minsky and Papert subjected locally conjunctive predicates on the set of images to thorough analysis and showed their limited possibilities. Two-dimensional grammars have enormously wider possibilities. Below it will be shown that two-dimensional grammars are a universal means for prescribing sets of feasible images, i.e., that for any finite set of images a grammar can be constructed, generating images of this set, and only them.

The differences between two-dimensional grammars and the predicates of Minsky and Papert can be illustrated by the following two examples. These are just the examples on which Minsky and Papert underscored the limited possibilities of local predicates on the set of images.

THE "PARITY" PREDICATE

The "parity" predicate gives a set of images, i.e., a set of sequences $v(t_1), v(t_2), \ldots, v(t_m)$, where t_i is an element of the visual field; v is a variable, taking on the values 0 and 1, where the number of signals, taking the value 1, is even. This predicate does not have any practical value, and an algorithm for testing the property of "parity" can be easily imagined. Nevertheless, it is interesting in that it cannot be expressed by any locally

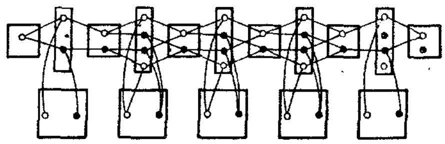

Fig. 1

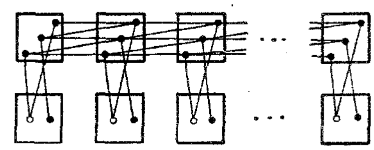

Fig. 2

conjunctive predicate, except if among the partial predicates m_i there is at least one predicate that depends on all of the signals, composing the image [30].

Figure 1 represents a grammar generating this set of images. The grammar is given in the form of a graph whose vertices correspond to the elements of the set σ (called above nodes); the arcs connect the vertices σ and σ', lying in adjacent cells and lead to the set $\overline{\sigma\sigma}(t(\sigma), t(\sigma'))$, i.e., form a resolved pair. The lower row of squares represents the visual field and the upper row of squares and rectangles, the description field.

A black circle in a cell of the visual field corresponds to one and a white circle, to zero. A black (white) circle in a cell of the description field designated by a square means that the sequence of signals in the cells of the visual field from the left to this square contains an odd (even) number of ones.

Every vertex in a cell of the description field, designated by a rectangle, is connected by an arc to only one vertex in each of the adjacent cells. These connections express the rule of alternation of the property of even – odd parity when adjoining the current symbol to the portion of the sequence. Namely, if this current symbol is equal to one, then the odd – even parity property changes; otherwise, it is conserved.

By the example of this simple grammar we have aimed to underscore the difference between signals and structural elements. If an image is given, this means that the value of the signal is given in each cell of the visual field. In this sense it can be said that the signal enters directly into the image. Prescription of an image, however, is not at all necessarily equivalent to the unique prescription of its description. The grammar G defines only the set of descriptions, each of which associated to a given image, forms a feasible variant. In the particular case this set of descriptions may contain only a single description. But even in this case the value of the description in a given cell is not defined by the value of the image at any one cell of the visual field or even by some small group of values. In the present example, for example, to define for any one cell the value of the feasible description of the given image, it is necessary to know the entire image.

THE "CONNECTIVITY" PREDICATE

The concept of connectivity is an important concept of visual analysis. Aside from this, this predicate cannot be expressed by means of any locally conjunctive predicate.

For simplicity, let us consider the one-dimensional case.

Let the image be represented in the form of a sequence v(1), v(2), ..., v(n), each of whose elements may be equal to zero or one. The image is assumed to satisfy the "connectivity" predicate if it can be represented in the form

$$\underbrace{000\ldots0}_{i_1 \text{ times}}\ \underbrace{111\ldots1}_{i_2 \text{ times}}\ \underbrace{000\ldots0}_{i_3 \text{ times}},\quad i_1+i_2+i_3=n.$$

A grammar giving a predicate of this kind has a simple form (Fig. 2). In the graph represented in Fig. 2, as before, the lower row of squares corresponds to the visual field and the upper, to the description field. A black circle in a visual field cell corresponds to the signal value 1 and a white one, to the value 0.

UNIVERSALITY OF TWO-DIMENSIONAL GRAMMARS

It is more or less obvious that two-dimensional grammars are a universal means of prescribing arbitrary sets of images if only the visual field Tv is finite.

Indeed, let there be given a set of images $B=\{\mathbf{v}_j, j\in J\}$, where J is a certain set of indices. This set is finite if the set B does not contain two identical images, since the visual field Tv and the alphabet V are finite. The grammar generating images in the set B and only these has the following form.

The description field is taken isomorphic to the visual field, where the cells corresponding in this isomorphism are adjacent. In the description field the adjacent predicate can be arbitrary, as long as the field remains simply connected. The alphabet of structural elements must be taken equal to $S = J \times Tv$; the set $ZZ(t, t')$ for $t \in Ts$ and $t' \in Ts$ is equal to $\{((j,t)(j,t'));\ j \in J\}$; the set $ZZ(t, t')$ for $t \in Ts,\ t' \in Tv$ is equal to $\{((j,t), v_j(t'));\ j \in J\}$.

The proof of universality of two-dimensional grammars signifies that in the worst case two-dimensional grammars always have the possibility of simply storing all feasible images.

We would like to state that for a given set of visual signals two-dimensional grammars are not only theoretically, but also practically, applicable. But this cannot be done, since the concept of "visual image" is not a mathematical concept. However, it is possible to confirm this fact by the series of examples presented below, each of which illustrates some procedure for the construction of two-dimensional grammars for prescribing sets of images that are of interest from the practical viewpoint. Furthermore, to each of the examples there corresponds a sometimes far from simple recognition problem that consists in finding an image, feasible in a given grammar, most similar to the image presented for recognition.

In the consideration of the examples of construction of grammars, we shall not touch on the question of recognizing the images generated by them. This question will be the subject of later sections, in which a single algorithm will be shown for solving an arbitrary recognition problem, provided it is formulated in terms of two-dimensional grammars.

EXAMPLES OF TWO-DIMENSIONAL GRAMMARS

1. In Martelli's work [31], in which an edge-detection algorithm is proposed, the following set of images is considered. Any image in this set defines two subsets in the visual field: the subset of white cells and the subset of black cells. A black cell having a white cell in its neighborhood is called an edge cell. An arbitrary row of cells must contain not more than one edge cell. For example, the images in Figs. 3 and 4 are permissible, and those of Figs. 5 and 6 are impermissible.

In order that the further exposition not be cumbersome, we shall introduce a further restriction, namely, a white cell must be found to the left of an edge cell.

A grammar generating this set of images is given in Fig. 7 in graph form. The ensemble of large cells composes the description field and the ensemble of small cells, the visual field. The alphabet of structural elements consists of three symbols that can be called "white," "edge," and "black," represented in the form of graph vertices, located in the upper left-hand corner of the cell, in its center, and in its lower right-hand corner, respectively. The signal alphabet consists of two elements – 1 (denoted by a black circle) and 0 (denoted by a white circle). Permissible pairs of symbols are selected in the following way.

In a certain cell of the visual field a 0-signal is possible only when the structural element "white" has been selected in the corresponding cell of the description field. Otherwise, only the 1-signal is possible.

The connections between two horizontally adjacent cells of the description field are made in such a way that to the right of the "edge" symbol only the "black" symbol is possible, and to the left only the "white" symbol. At the same time to the right of the "black" symbol only a "black" symbol is possible, and to the left of a "white" one only a "white" one.

The connections between vertically adjacent cells in the description field are made in such a way that adjacency of a "black" cell with a "white" one is excluded, but any other combinations are admitted.

Let us show that the image in Fig. 6 is unfeasible. Since the signal in cell t_1 is equal to 1, the description of this image in the corresponding cell of the description field must contain either the symbol "edge" or the symbol "black." In either of these two cases in the cell t_2, according to the feasible horizontal couplings, there can only be a "black" symbol. In the cell t_3 of the description field only a "white" symbol is possible, since the cell adjacent to it in the visual field contains the 0-signal. However, the "black – white" pair is not admitted by the constraints adopted for adjacent black symbols. Consequently, no structural element can be written into cell t_3, and the image presented does not have a feasible description and thus is not feasible.

This grammar is simple in the sense that the storage of the graph prescribing this grammar requires a small volume of memory. At the same time it prescribes an extensive set, containing approximately $m \cdot 3^n$ images, where m and n are the horizontal and vertical dimensions of the visual field.

2. Let us consider the grammar forming the concept "curvilinear strip of constant width." Every image, as in the preceding example, divides the visual field into two subsets – black cells and white cells. As before, a black cell adjacent to a white one is called an edge cell. The following constraints are imposed on the images:

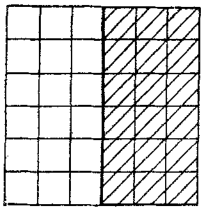

Fig. 3

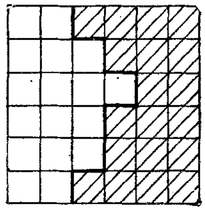

Fig. 4

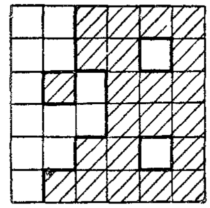

Fig. 5

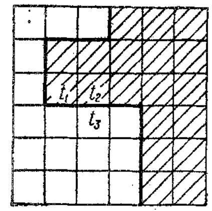

Fig. 6

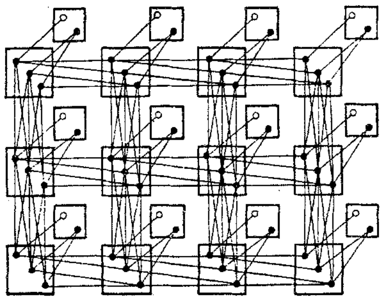

Fig. 7

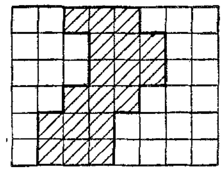

Fig. 8

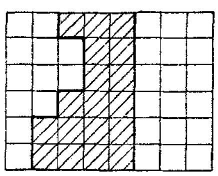

Fig. 9

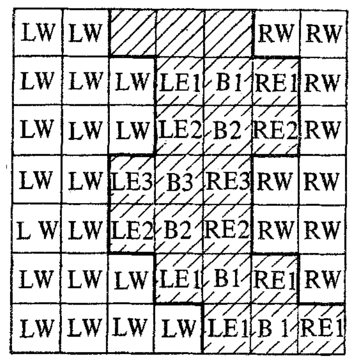

Fig. 10

only two edge cells are possible in any one row;
only black cells occur between the two edge cells;
the number of black cells in a row can be arbitrary (not less than three), but the same for all rows.

Examples of feasible and unfeasible images are shown in Figs. 8 and 9.

The feasible images considered are such that the configuration of the right edge of the strip must exactly repeat the configuration of the left edge. This means that in the image there are rigidly coupled sections that are geometrically substantially distant from each other. It can be found that such constraints, namely, constraints connecting remote sections of the visual field, are hard to define by means of those essentially local constraints that are available to two-dimensional grammars. Let us show that in the present example this is not the case.

The visual field, the description field, and the adjacency predicates are given in exactly the same way as in the previous example. The set of structural elements consists of the symbols: LW ("left white"); LE1, LE2, LE3 ("left edge" of three types), B1, B2, B3 ("black" of three types), RE1, RE2, RE3 ("right edge" of three types), RW ("right white").

The set of signals is as follows: w ("white"), b ("black").

The sense of the structured elements is illustrated in Fig. 10, where a feasible image and its feasible description are given.

The set Z(t, t'), where t is a cell of the description field, and t' the cell adjacent to it in the visual field, are given so that for z(t') = w the symbol z(t) can take on only the values LW and RW, while for z(t') = b, all the remaining values.

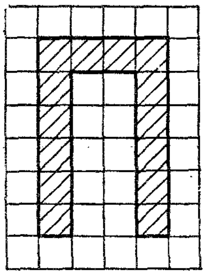

Fig. 11

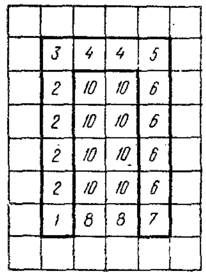

Fig. 12

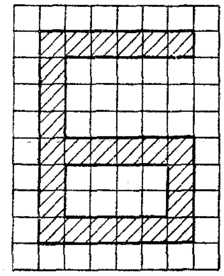

Fig. 13

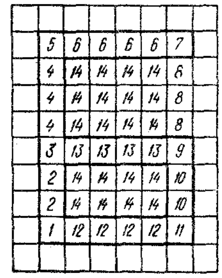

Fig. 14

The set Z(t, t'), where t and t' are vertically adjacent cells in the description field, must be given so that:

above a structural element RE1 only a structural element RW be authorized, and above a structural element LE1, only the structural elements B1, B2, B3;
above a structural element LE2 only elements LE1, LE2, LE3 be authorized, and above a structural element RE2, the elements RE1, RE2, RE3;
above a structural element LE3 only an element LW be authorized and above a structural element RE3, the structural elements B1, B2, B3.

The set Z(t, t') for two horizontally adjacent cells of the description field is given in such a way that in a row where an element LE1 (or LE2, LE3) is found, after a series of elements B1 (or B2, B3) there is authorized only an element RE1 (or RE2, RE3).

The introduction of these constraints is equivalent to the assignment of the following meaning to the structural elements. The symbols LE1 and RE1 (or LE3 and RE3) signify edge points for which in the transition to the next row above the edge shifts one cell to the left (or to the right). The symbols LE2 and RE2 signify edge points such that in the transition to the next row above the horizontal coordinate does not change.

Thus, the indices 1, 2, 3 of the structural elements LE, RE contain information about the direction to which the edge shifts in the passage from row to row. Just this information is transmitted from the left edge of the strip to the right by the index of the structural element B.

3. There exists a class of recognition problems requiring the use of a special variant of two-dimensional grammars, which it is appropriate to call three-dimensional grammars. These problems consist in the recognition of moving objects, i.e., in the simultaneous analysis of a sequence of images, representing photographs of the objects at successive instants of time. Constraints are imposed on each individual image, taking into account its inner structure and expressed by a two-dimensional grammar. Aside from these constraints there exist further constraints on the pairs of consecutive images in time, due, for example, to the inertia of the photographed object itself. It is to be understood that in this case the decision as to the state of the photographed object at the i-th instant of time must not be taken only on the basis of the image observed at this instant. By virtue of the inertia of the object the information about its state at a fixed instant of time is also contained in its images taken at other instants of time.

As an example we shall consider the problem of recognizing a moving edge.

Let it be known that the image of the edge at any instant of time satisfies the constraints considered in example 1. Assume that it is also known that the position of the edge in time cannot change rapidly. We shall prescribe such a set of feasible sequences of images by means of the following grammar.

The visual field Tv and the description field Ts will be given in the form of the union $Tv^i = \bigcup_i Tv^i, Ts = \bigcup Ts^i$, where the index i signifies the time. The field T will be given in the form of the union $T = \bigcup_i T^i$, where $T^i = Tv^i \bigcup Ts^i$. The elements of the field will be denoted by the pair t = (τ, i). The adjacency predicate for cells $t = (\tau, i)$ and $t' = (\tau', i)$, i.e., cells with identical index i, will be given in the same way as in the first example. Also as in this example the sets ZZ(t, t') will be given, if only the cells t and t' belong to the same field T^i. The introduction of these constraints leads to the image v^i for an arbitrary time i, i.e., the mapping $Tv^i \to V$, coinciding with one of the feasible images considered in example 1. In order to introduce constraints on the pairs of consecutive images in time, it is necessary to declare the cells t = (τ, i) and t' = (τ, i + 1) adjacent if these cells belong to the description field and not the visual field. For these pairs of cells the sets ZZ(t, t') are taken in such a way as to exclude the possibility of adjacency (in time) of a "black" symbol with a "white" one. This constraint leads to the situation where if for some time i a cell of the visual field (τ, i) were,

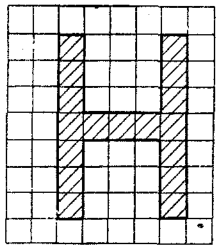

Fig. 15

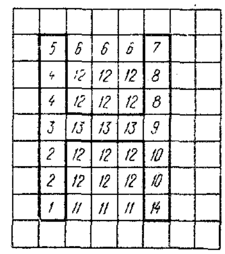

Fig. 16

for example, white, then at the time i + 1 the cell (τ, i + 1) can either remain white or become an edge cell. The grammar thus constructed admits only such sequences of edge images for which the edge during one time interval can shift by not more than one cell.

4. In this example is illustrated the idea of constructing two-dimensional grammars for certain classes of stylized handwritten characters. All of these grammars will be given on the same field as in example 1, and with the same adjacency predicate. In Figs. 11 and 12, by means of the pair "image – description," is illustrated the idea of constructing a grammar to generate images of the letter П. In the figures it is evident that the structural elements 1-7 correspond to the signal value ▨ and the elements 8 and 10, to the signal □. In Fig. 12 it is evident that the set of feasible horizontally adjacent pairs of structural elements consists of the pairs (□, 1), (□, 2), (□, 3), (□, □), (1, 8), (2, 10), (3, 4), (4, 4), (10, 10), (8, 8), (4, 5), (10, 6), (8, 7), (5, □), (6, □), (7, □) and does not contain, for example, the pairs (2, 6). It is also evident which pairs of vertically adjacent symbols are feasible. In an analogous way, Figs. 13 and 14 illustrate a grammar generating the set of images of the letter Б, and Figs. 15 and 16 the grammar of the letter H. Attention should be directed to the fact that the structural elements not only characterize those cells of the visual field in which the character strokes appear, i.e., not only the black cells, but also the white cells not occupied by strokes. Thus, to prescribe the grammar of the letter П it is necessary to distinguish three types of white cells – simple white cells (the symbol □), a white cell within the letter (the symbol 10), and the white cells between the lower ends of the vertical strokes. In example 2 it was already necessary to distinguish white cells of different types. However, here the different types of these cells have a more clearly expressed functional role. Namely, the presence of a structural element 10, different from the symbol □, excludes the possibility of constructing inside one letter П another letter of smaller size. The presence of the structural element 8 excludes the possibility of constructing a letter П in which the vertical strokes would not be the same length.

5. In these examples it was assumed that the signals from which the image is composed give exhaustive information as to whether a cell is black or white. In a number of cases it is technically possible only to measure the amount of light reflected from a given cell of the visual field. Different causes, for example, varying the conditions of illumination of the image, lead to situations in which in some images cells must be considered to be black that reflect more light than cells that in other images must be considered to be white. Nonuniform illumination over the visual field can even lead to the situation where such a pair of cells even appears within the limits of a single image.

Below we consider grammars that can be called multilevel two-dimensional grammars. They are convenient for prescribing such sets of images that are not composed of "black" and "white" signals, but of signals, for example, 0, 1, 2, ..., 9, 10, representing the quantity of reflected light.

Let it be known, for example, that in black cells of the image the quantity of reflected light can vary from 0 to 6, and in white cells from 4 to 10. These values are not independent. The dependency between them can be expressed, for example, in that:

if two adjacent cells of the visual field are both black or both white, then the quantity of light reflected from these cells may differ by not more than 1;
if of two adjacent cells one is white and the other black, then the quantity of reflected light from the former must be two units greater than from the latter.

Let the grammar G_1 generate the set of images composed of the symbols "black" and "white." By G_2 we denote a grammar generating images composed of the values of reflected light in the way stated above. The field of this grammar consists, as before, of the visual field Tv and the description field Ts. The latter consists of three subfields, called in the sequel levels. The first level of grammar G_2 is the description field of grammar G_1 and the second level, the visual field of G_1. The adjacency predicate in these two levels, as well as the sets ZZ(t, t') for pairs of adjacent cells are given in the same way as in grammar G_1. By virtue of this, in the second level of grammar G_2 there can arise only such configurations as are feasible images in grammar G_1. The third level of

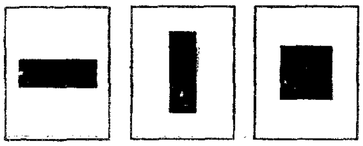

Fig. 17

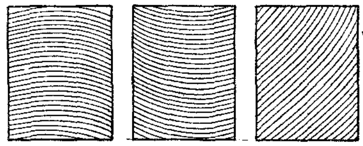

Fig. 18

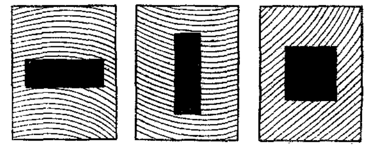

Fig. 19

grammar G_2 is a set of cells isomorphic to the visual field of grammar G_1 and, consequently, to the second level of grammar G_2. In each cell of this level may be placed a structural element from the alphabet $\{0b, 1b, 2b, 3b, 4b, 5b, 6b, 4w, 5w, 6w, 7w, 8w, 9w, 10w\}$. Each value of this structural element contains information both about the possible quantity of reflected light and whether the given cell of the visual field is black or white. The cell t of the third level is adjacent to a cell t' of the second level if the cells t and t' correspond in the adopted isomorphism. The sets ZZ(t, t') for these pairs of cells are selected in such a way as to satisfy the above constraints, defining the possible quantity of reflected light as a function of whether the cell is white or black. Specifically, if in the cell t of the second level is placed a "black" symbol, then in the corresponding cell t' of the third level there can occur only one of the symbols 0b, 1b, 2b, 3b, 5b, 6b. For z(t) = "white" z(t') can only take on a value 4w, 5w, 6w, 7w, 8w, 9w, 10w. If t and t' are adjacent cells, where both belong to the third level, then feasible pairs of symbols (z(t), z(t')) are selected according to the above conditions, which relate the quantities of reflected light in adjacent cells.

The fourth level of grammar G_2 is the visual field of this grammar, isomorphic to the visual field of grammar G_1, and thus to the third level of grammar G_2. In the visual field of grammar G_2 a simple rewriting of the symbols of the third level takes place, omitting the symbols b and w. This underscores the fact that in the signals composing the image itself information is already absent that some given cell of the visual field is white or black, and only information is present about the quantity of reflected light.

6. Interest is presented by a grammar of the type "image on a background." Let the grammar G_1 generate a certain set of images, for example, those presented in Fig. 17, and grammar G_2 the set of images, for example, given in Fig. 18. Using these two grammars, it is required to construct a grammar G_3, generating the former images on the background of the latter, as shown, for example, in Fig. 19.

In the simplest case the process of generating the image v, representing the image v_1 on the background of image v_2, may be represented in the form of a position-by-position operation, analogous to disjunction. This means that a certain cell in image v must be white only when it is white both in image v_1 and in image v_2. The grammar G_3, generating such a set of images, can be obtained in the following way.

Let T^i, Ts^i, Tv^i $(i = 1, 2)$ be the field, the description field, and the visual field of the grammar G_i, respectively. In this case the description field of grammar G_3 must be the union of fields T^1, T^2, and a certain field T^{12}, isomorphic both to the visual field Tv^1 and the visual field Tv^2. The field T^1 of grammar G_1 (and, consequently, the visual field and the description field of this grammar) is a subset in the description field of grammar G_3. The adjacency predicate and the set ZZ(t, t') for this subset are chosen as in the grammar G_1. Therefore, on the subset Tv^1, constituting the visual field of grammar G_1, but a portion of the description field of grammar G_3, only such configurations are feasible that are feasible images in G_1. In an analogous way it is obtained that in the subset Tv^2, representing the other part of the description field of grammar G_3, only those configurations are feasible that are feasible images in the grammar G_2.

In a cell of field T^{12} are written symbols from the alphabet (bb, bw, wb, ww). Let $t^1 \in Tv^1$, $t^2 \in Tv^2$, $t^{12} \in T^{12}$ and let these cells correspond in the adopted isomorphism. The sets $ZZ(t^{12}, t^1)$ and the sets $ZZ(t^{12}, t^2)$ must be given in such a way that the symbols $z(t^{12})$ contain information both about $z(t^1)$ and about $z(t^2)$. For example, $ZZ(t^{12}, t^1) = \{(bb, b), (bw, b), (wb, w), (ww, w)\}$ for all $t^{12} \in T^{12}, t^1 \in Tv^1$, $R(t^{12}, t^1) = 1$; $ZZ(t^{12}, t^2) = \{(bb, b), (bw, w), (wb, b), (ww, w)\}$ for all $t^{12} \in T^{12}$, $t^2 \in Tv^2$, $R(t^{12}, t^2) = 1$.

The visual field Tv of grammar G_3 is isomorphic to the field T^{12}. In a certain cell $t \in Tv$ there may be written the symbol w only when in the corresponding cell of field T^{12} is written the symbol ww. In all other cases the symbol b must be written into the cell.

7. In this example the situation considered in the preceding example will be generalized.

Let $v_1, v_2, \ldots, v_n$ be feasible images in the grammars $G_1, G_2, \ldots, G_n$, respectively. It is required to construct a grammar G, generating an image v, that can be obtained from the images $v_1, v_2, \ldots, v_n$ by means of a given cellular operation *f*. This means that the value of the image v in the cell t is defined by the values of the image $v_1, v_2, \ldots, v_n$ only in the cell t: $v(t) = f(v_1(t), v_2(t), \ldots, v_n(t))$.

Aside from the example 6 it is possible to indicate many other examples where such a representation of the set of feasible images can be useful. An interesting case is where $v_n(t)$ takes on the values 1, 2, ..., n − 1, while the function f takes the form $v(t) = v_{v_{n(t)}}(t)$. Such a function holds when the image v in the visual field is a compilation of several images, each of which occupies its own portion of the visual field and is generated by its own grammar. At the same time, the division of the visual field into parts, i.e., domains of effect of one or another grammar, is not arbitrary, but is given by the grammar G_n.

Consider a procedure that allows a grammar G to be constructed, generating an image v represented in the form $v(t) = f(v_1(t), v_2(t), \ldots, v_n(t))$, where v_i is the image generated by a certain grammar G_i.

This procedure is very similar to that considered in the preceding example. It consists in representing the description field Ts of the grammar G in the form of the union of fields T^i of the grammars G^i and a field T*. The field T* is isomorphic to the visual field. Into its cells are written symbols in the alphabet $V_1 \times V_2 \times \ldots \times V_n$, where V_i is the alphabet of symbols of the grammar G_i. The contents of the cell of field T* must, in correspondence with the given function f, uniquely define the contents of the corresponding cell of the visual field.

RECOGNITION OF IDEAL IMAGES. PERFECT GRAMMARS

Let there be given a grammar G, i.e., an alphabet $Z = V \cup S$, a field $T = Tv \cup Ts$, an adjacency predicate $R: T \times T \to \{1, 0\}$, and a set ZZ(t, t') for arbitrary t and t' such that R(t, t') = 1. In the sequel we shall call the pair (T, R) a structure of the grammar. The problem of recognition of ideal images consists in deciding for a given image v its feasibility in a given grammar or, in a somewhat different formulation, it is required to give a reply to the question of the existence of a description s such that the pair (v, s) is a feasible variant.

Below, a single algorithm is proposed that for arbitrary pair (G, v) "grammar – image" will give one of the following replies: "feasible," "unfeasible," "don't know" – whose meanings are obvious.

The basic elementary operation of this algorithm is the operation called elimination. The elimination operation is applied to vertices, i.e., elements of the sets $\bar{\sigma}(t)$, $t \in T$, and to arcs, i.e., to elements of the sets $\overline{\sigma\sigma}(t, t')$, $(t, t') \in N$. Initially, all of these elements are considered uneliminated. The elimination operation is only applicable to elements that have not been eliminated, but which as a result of this operation become eliminated.

The initial set of eliminations consists in eliminating for every cell t of the visual field Tv all the vertices in the set $\sigma(t)$ except the vertex σ^*, such that $z(\sigma^*) = v(t)$, where v(t) is the value of the unknown image in the cell t.

The basic set of eliminations consists in the following. For every pair of adjacent cells t and t' $(t \in T, t' \in T)$, $(t, t') \in N$, the arcs $(\sigma, \sigma') \in \overline{\sigma\sigma}(t, t')$ are eliminated when at least one of the following four conditions is satisfied: 1) $\sigma \bar{\in} \bar{\sigma}(t)$; 2) $\sigma' \bar{\in} \bar{\sigma}(t')$; 3) σ has been eliminated; 4) σ' has been eliminated.

Aside from this, for every cell t ∈ T the vertex $\sigma \in \bar{\sigma}(t)$ is eliminated if there exists a cell t', adjacent to the set t, such that all of the arcs of the form $(\sigma, \sigma') \in \overline{\sigma\sigma}(t, t')$ have been eliminated.

The algorithm ceases to operate when no elimination can be effected for the above rules. Since the number of eliminations is finite, the algorithm ceases to operate in a finite number of steps.

The following two assertions are of obvious validity:

if in the course of operation of the algorithm for at least one cell t ∈ T all of the vertices $\sigma \in \bar{\sigma}(t)$ have been eliminated, then the unknown image has not been generated by the given grammar;

if after termination of the operation of the algorithm in each of the sets $\bar{\sigma}(t)$, $t \in T$, only one vertex is found to be uneliminated, then the unknown image is generated by the given grammar.

The algorithm can, however, terminate operation without satisfaction of these two conditions. For example, it can happen that after termination of the operation, for some t, not one but several elements in the sets $\sigma(t)$ are uneliminated. In this case, in general, it is not possible to give an answer to the question of the feasibility of the unknown image, and additional investigation is required. It is possible, however, to bring into consideration a class of grammars in which from the fact that after termination of the algorithm there are vertices that have not been eliminated in any of the sets $\bar{\sigma}(t)$, $t \in T$, it follows that the unknown image v is feasible. We call the grammars of this class perfect.

Definition 1. A grammar is called undiminishable if the operation of elimination cannot be applied to it. The grammar given by the field T', the predicate R', and the sets Z'(t) and ZZ'(t, t'), is called a subgrammar of the grammar given by the field T, the predicate R, and the sets Z(t), ZZ(t, t'), if $T' = T$, $R' = R$, while $Z'(t) \subset Z(t)$ for all $t \in T$; $ZZ'(t, t') \subset ZZ(t, t')$ for all $(t, t') \in N$. A grammar is contradictory if the set of feasible images generated by it is empty. A grammar is called nonempty if for any cell $t \in T$ the set of vertices $\bar{\sigma}(t)$ is not empty. A grammar is called perfect if any of its nonempty undiminishable subgrammars is not contradictory.

The class of perfect grammars is complete in the sense that for any two-dimensional grammar there exists a perfect grammar, generating the same set of feasible images. The proof of this fact is analogous to that presented earlier of the universality of grammars and is not presented here.

Interest is also presented by structures (T, R) such that any grammar over the structure is perfect, regardless of the sets Z(t) and ZZ(t, t'). An example of such a structure is an arborescent structure.

Definition 2. We shall assume that the field T and the predicate R admit a cycle of length greater than two if there exists at least one sequence of cells $t_0, t_1, \ldots, t_n$, $n > 2$, such that $t_0 = t_n$, $t_{i+1} \neq t_{i-1}$ for all i = 1, 2, ..., n − 1 and $R(t_i, t_{i+1}) = 1$ for all i = 0, 1, 2, ..., n − 1. If the structure (T, R) is simply connected and does not admit any cycle of length greater than two, we shall call it arborescent.

Assertion. Let a grammar G be given by a field T, a predicate R, an alphabet Z, and sets ZZ(t, t') for (t, t') ∈ N. Let the structure (T, R) be arborescent. In this case the grammar G is perfect.

Proof. Consider the nonempty and undiminishable subgrammar G' of the arborescent grammar G. The grammar G' is also arborescent, since by definition a subgrammar is given over the same pair (T, R) as the grammar. Let us take a certain cell t_0 and call it a cell of zero rank. We take any cell t* and consider the sequence of cells t_0, t_1, t_2, . . . , t^*, beginning with the cell t_0, terminating with the cell t*, and such that no one cell in it is met twice, and any two cells t_i, t_{i-1} $(i = 1, 2, \ldots)$ in it are adjacent. It is obvious that such a sequence exists; otherwise, the given structure would not be simply connected; furthermore, this sequence for an arbitrary pair of cells t_0 and t* is uniquely defined; otherwise, the structure would not be arborescent. We denote the length of this sequence by n + 1, where n will be called the rank of the cell t* with respect to the cell t_0, whose rank is zero.

In accordance with the arborescence of the structure the following are valid:

a) no two cells of the same rank are adjacent;
b) for every cell of rank i there exists a unique cell adjacent to it of rank i − 1;
c) no two cells whose ranks differ by more than unity are adjacent.

By virtue of this it is possible to indicate the following procedure for finding a feasible variant in an undiminishable arborescent grammar.

1. The zero stage. In a cell t_0 of zero rank we select an arbitrary vertex in the set $\bar{\sigma}(t_0)$. This can be done since the subgrammar G' is not empty.

2. The i-th stage begins when in some cell with rank less than i an element z(t) is chosen in such a way that for every pair of adjacent cells t and t', of rank less than i, the pair z(t) and z(t'), chosen in these cells, is feasible. Examining the cells of rank i in any sequence, for each such cell t we find the cell t' of rank i − 1, adjacent to it (and such a cell is unique), and then in the cell t we take a vertex σ(t) such that the pair $(\sigma(t), \sigma(t')) \in \overline{\sigma\sigma}(t, t')$. Such a vertex necessarily exists, since otherwise the subgrammar G' would not be undiminishable, since from the set $\bar{\sigma}(t')$ the vertex $\sigma(t')$ would be eliminated. We note that in the choice of vertex in the cell of i-th rank it is not necessary to consider which vertices have been selected in the other cells of this rank, since no two cells of i-th rank are adjacent. Furthermore, in the choice of vertex in the cell of i-th rank it is not necessary to consider which vertices have been selected in cells of rank less than i − 1, since these cells are also not adjacent to cells of rank i.

Successively applying the i-th stage for i = 1, 2, ..., n, where n is the maximum rank, we obtain a variant feasible in the grammar G.

We have shown that any undiminishable subgrammar of an arborescent grammar generates at least one feasible variant and thus have completed the proof of the assertion.

Thus, an algorithm based on elimination can be applied to the solution of problems in the membership of images in given two-dimensional languages. For an arbitrary grammar, however, it is not excluded that the algorithm may give the reply "don't know." In the case of special grammars, called perfect, the reply "don't know" is excluded.

THE RECOGNITION OF REAL IMAGES

The problem of recognizing real images is formulated in the following way.

Let there be given a grammar G, generating ideal images **v** in the visual field Tv, and a real image **x**, i.e., a mapping of the visual field Tv onto a certain alphabet X. Let there be given a similarity function F(**v**, **x**), having the form $F(\mathbf{v}, \mathbf{x}) = \sum_{t \in Tv} f(v(t), \mathbf{x})$. For these initial data it is required to find the ideal image corresponding to the maximum similarity with the given real image.

Below we propose an algorithm solving this problem similarly to the way the above algorithm solved the problem of recognizing ideal images. Namely, this algorithm either indicates a certain ideal image, and in this case this indicated image is optimal, or it replies "don't know." It is essential for it that the indication of ideal images that are not optimal is excluded. No less important is also the existence of perfect grammars for which, as will be shown below, the reply "don't know" is excluded.

To formulate the algorithm, and for its subsequent justification, we shall introduce additional concepts.

We denote by M(t) the set of cells adjacent to the cell t, i.e., $M(t) = \{\tau : \tau \in T;\ R(t, \tau) = 1\}$. We divide the set M(t) into two disjoint subsets $M^+(t)$ and $M^-(t)$, satisfying the following condition: if $\tau \in M^+(t)$, then $t \in M^-(\tau)$, and *vice versa*, if $\tau \in M^-(t)$, then $t \in M^+(\tau)$. We associate to each pair (σ, τ), $(\sigma \in \bar{\sigma}, \tau \in M(t(\sigma))$ the real number φ_σ^τ, called a potential. We associate to every vertex σ the number $\Delta(\sigma) = \sum_{\tau \in M^+(t(\sigma))} \varphi_\sigma^\tau - \sum_{\tau \in M^-(t(\sigma))} \varphi_\sigma^\tau$, called the tension at the vertex σ. We associate to every pair of vertices (σ, σ'), connected by an arc, i.e., such that $(t(\sigma), t(\sigma') \in N)$, $(\sigma, \sigma') \in \overline{\sigma\sigma}(t(\sigma), t(\sigma'))$, the number $\delta(\sigma, \sigma')$, called the tension between the vertices σ and σ' or the tension on the arc (σ, σ'). The tension $\delta(\sigma, \sigma')$ is defined by the expression $\delta(\sigma, \sigma') = \varphi_\sigma^{t(\sigma')} - \varphi_{\sigma'}^{t(\sigma)}$, if $t(\sigma) \in M^+(t(\sigma'))$; $\delta(\sigma, \sigma') = \varphi_{\sigma'}^{t(\sigma)} - \varphi_\sigma^{t(\sigma')}$, if $t(\sigma') \in M^+(t(\sigma))$.

By definition, it is clear that $\delta(\sigma, \sigma') = \delta(\sigma', \sigma)$.

We designate by height of the vertex σ the quantity h(σ), defined by the formula $h(\sigma) = f(z(\sigma), \mathbf{x}) - \Delta(\sigma)$ if $t(\sigma) \in Tv$; $h(\sigma) = -\Delta(\sigma)$ if $t(\sigma) \in Ts$.

We designate by potential function the function

$$K(\{\varphi\}, \mathbf{x}) = \sum_{t \in Tv} \max_{\sigma \in \bar{\sigma}(t)} h(\sigma).$$

We designate by feasible vector $\{\varphi\}$ of potentials φ_σ^τ the vector satisfying the conditions $\delta(\sigma, \sigma') \geq 0$ for all pairs σ, σ' such that $(z(\sigma), z(\sigma')) \in ZZ(t(\sigma), t(\sigma'))$, $R(t(\sigma), t(\sigma')) = 1$; furthermore, h($\sigma$) $\leqslant$ 0 for all σ such that t(σ) $\in$ Ts.

The connection between potential function and similarity function is expressed by the following theorem.

THEOREM 1. Let $\{\varphi\}$ be an arbitrary vector of potentials, feasible in the grammar G; **v** is an arbitrary image generated by this grammar; **x** is an arbitrary real image. In this case $K(\{\varphi\}, \mathbf{x}) \geq F(\mathbf{v}, \mathbf{x})$.

Proof. The inequality

$$K(\{\varphi\}, \mathbf{x}) \geq \sum_{t \in Tv} \max_{\sigma \in \bar{\sigma}(t)} h(\sigma) + \sum_{t \in Ts} \max_{\sigma \in \bar{\sigma}(t)} h(\sigma) \quad (1)$$

holds, since the first sum on the right side by definition is equal to the potential function, while the second sum is nonpositive by virtue of the feasibility of the vector $\{\varphi\} : h(\sigma) \leqslant 0$ for all σ such that $t(\sigma) \in Ts$.

Let v* be a certain feasible image, s* its description, and z* a feasible variant (v*, s*). We shall say that a certain vertex σ' enters into the variant z* if $z^*(t') = z(\sigma')$. By analogy, a certain arc (σ', σ'') enters into the variant **z*** if $R(t(\sigma'), t(\sigma'')) = 1$ and further $z^*(t') = z(\sigma')$, $z^*(t'') = z(\sigma'')$.

The set of vertices σ entering into the variant z* is denoted by $B\sigma(z^*)$ and the set of arcs (σ',σ''), entering into z*, by $B\sigma\sigma(z^*)$.

The equality

$$\sum_{\sigma\in B\sigma(z^*)} \Delta(\sigma) + \sum_{(\sigma,\sigma')\in B\sigma\sigma(z^*)} \delta(\sigma,\sigma') = 0 \quad (2)$$

holds, since any potential φ_σ^τ either enters into both sums in this expression (if $\sigma\in B\sigma(z^*)$) or does not enter either of them (if $\sigma\,\bar{\in}\, B\sigma(z^*)$); in this if in one of these sums a given potential enters with the "+" sign, then it necessarily enters into the other with the "–" sign.

The second term in (2) is not negative by virtue of the feasibility of the vector $\{\varphi\}$. Consequently, the first term in (2) is not positive:

$$\sum_{\sigma\in B\sigma(z^*)} \Delta(\sigma) \leqslant 0. \quad (3)$$

From this it follows directly that

$$\sum_{t\in T_v} f(v^*(t), x) - \sum_{\sigma\in B\sigma(z^*)} \Delta(\sigma) \geqslant F(v^*, x). \quad (4)$$

The left side of inequality (4) is the sum of heights of the vertices σ, entering into the variant z*, i.e., is equal to $\sum_{\sigma\in B\sigma(z^*)} h(\sigma)$, as a result of which (4) can be rewritten in a somewhat different form:

$$\sum_{\sigma\in B\sigma(z^*)} h(\sigma) \geqslant F(v^*, x). \quad (5)$$

The inequality

$$\sum_{t\in T_v} \max_{\sigma\in\bar{\sigma}(t)} h(\sigma) \geqslant \sum_{\sigma\in B\sigma(z^*)} h(\sigma)$$

is obvious.

The left side of this inequality coincides with the right side of inequality (1), which allows us to write

$$K(\{\varphi\}, x) \geqslant \sum_{\sigma\in B\sigma(z^*)} h(\sigma). \quad (6)$$

The joint application of inequalities (6) and (5) leads to the inequality

$$K(\{\varphi\}, x) \geqslant F(v, x).$$

Q.E.D.

As soon as the potential function is an upper bound on the similarity function, interest is presented by the properties of those vectors of potentials that yield a minimum to the potential function.

To investigate these properties we shall introduce some further concepts.

Let there be given a grammar G and a vector $\{\varphi\}$ of potentials, feasible in this grammar. We shall denote by $G'(\{\varphi\})$ the grammar representing a subgrammar in G and obtained from it by eliminating from the sets Z(t), $t \in Tv$ those symbols z for which $h(z,t) \neq \max_{z\in Z(t)} h(z,t)$, eliminating from the set ZZ(t, t') those pairs (z, z') for which $\delta((z,t),\ (z',t')) \neq 0$, and also eliminating from the sets Z(t) ($t \in Ts$) those elements z for which $h(z, t) \neq 0$.

The properties of the vectors of potentials yielding a minimum potential function, are expressed by the following theorem.

THEOREM 2. If the vector $\{\varphi\}$, feasible in the grammar G, yields a minimum potential function, then the grammar $G'(\{\varphi\})$ contains an undiminishable subgrammar.

Proof. The problem of seeking a minimum potential function can be reduced to a linear programming problem in the following way. Let h(t) ($t \in Tv$) be a variable, called the height of the cell t and defined by the expression $h(t) = \max_{\sigma\in\bar{\sigma}(t)} h(\sigma)$. In this new notation the potential function is a linear form $\sum_{t\in T_v} h(t)$ that is to be minimized under the following linear constraints:

a) $h(t(\sigma)) \geqslant h(\sigma)$ for all $t(\sigma) \in Tv$;

b) $0 \geqslant h(\sigma)$ for all $\sigma \in \bar{\sigma}(t)$, $t \in Ts$;

c) $h(\sigma) = f(v(t), x) - \left(\sum_{t' \in M^+(t(\sigma))} \varphi_\sigma^{t'} - \sum_{t' \in M^-(t(\sigma))} \varphi_\sigma^{t'} \right)$ for all $t \in Tv$, $\sigma \in \bar{\sigma}(t)$;

d) $h(\sigma) = -\left(\sum_{t' \in M^+(t(\sigma))} \varphi_\sigma^{t'} - \sum_{t' \in M^-(t(\sigma))} \varphi_\sigma^{t'} \right)$ for all $t \in Ts$, $\sigma \in \bar{\sigma}(t)$;

e) $\varphi_\sigma^{t(\sigma')} - \varphi_{\sigma'}^{t(\sigma)} \geqslant 0$ for all pairs (σ, σ') such that $t(\sigma) \in M^+(t(\sigma'))$, $(\sigma\sigma') \in \overline{\sigma\sigma}(t(\sigma), t(\sigma'))$.

To this linear programming problem there corresponds the dual problem with variables $\alpha(\sigma)$, $(\sigma \in \bar{\sigma})$ and $\beta(\sigma, \sigma')$, $((\sigma, \sigma') \in \overline{\sigma\sigma}(t(\sigma), t(\sigma'))$, $(t(\sigma), t(\sigma')) \in N)$.

Let $\Omega(t', \sigma)$ be the set of vertices defined for arbitrary σ and arbitrary $t' \in M(t(\sigma))$, and consisting of those vertices $\sigma' \in \bar{\sigma}(t')$ that appear in the cell t', i.e., $\sigma' \in \bar{\sigma}(t')$, and that are connected by an arc to the vertex σ, i.e., $(\sigma, \sigma') \in \overline{\sigma\sigma}(t(\sigma), t(\sigma'))$. In this case, the constraints that the dual variables must satisfy, have the following form:

1) $\alpha(\sigma) = \sum_{\sigma' \in \Omega(t', \sigma)} \beta(\sigma, \sigma')$ for all $\sigma \in \bar{\sigma}$ and $t' \in M(t(\sigma))$;

2) $\alpha(\sigma) \geqslant 0$ for all $\sigma \in \bar{\sigma}$;

3) $\beta(\sigma, \sigma') \geqslant 0$ for all (σ, σ') such that $R(t(\sigma), t(\sigma')) = 1$, $(\sigma, \sigma') \in \overline{\sigma\sigma}(t(\sigma), t(\sigma'))$;

4) $\sum_{\sigma \in \bar{\sigma}(t)} \alpha(\sigma) = 1$ for all t ∈ T.

It should be noted that the variables $\alpha(\sigma)$, $t(\sigma) \in Tv$ of the dual problem correspond to the constraints $h(t(\sigma)) \geqslant h(\sigma)$ of the primal problem, the variables $\alpha(\sigma), t(\sigma) \in Ts$ correspond to the constraints $h(\sigma) \leqslant 0$ and, finally, the variables $\beta(\sigma, \sigma')$ correspond to the constraints e), namely, $\delta(\sigma, \sigma') \geqslant 0$.

One of the duality theorems [32] states that if the optimal vector $\{\varphi^*\}$ of the primal problem is such that certain constraints – inequalities – are strictly satisfied, then it is possible to equate to zero those dual variables that correspond to these constraints and at the same time the constraints of the dual problem do not become contradictory. We shall show that if for a certain $\{\varphi\}$ the grammar $G'(\{\varphi\})$ does not contain an undiminishable grammar, then the equation to zero of the dual variables, corresponding to the inequalities of the primal problem that are strictly satisfied leads to contradiction of the conditions of the dual problem.

The grammar $G'(\{\varphi\})$ is formed from the grammar G by elimination of those vertices σ or those arcs (σ, σ') whose weights $\alpha(\sigma)$ or $\beta(\sigma, \sigma')$ are equal to zero. Let us confirm that further application of the elimination rules to the grammar $G'(\{\varphi\})$ also leads to elimination of only those elements σ or pairs (σ, σ') whose weights are necessarily equal to zero, if only these weights do not contradict the conditions 1)–4).

Assume that before a certain elimination has been effected only those vertices or arcs have been eliminated whose weights must be equal to zero. Let in the current step a certain arc (σ, σ') be eliminated in connection with the previous elimination of the vertex σ. This means that the weight $\alpha(\sigma)$ of this vertex was equal to zero. But in such a case, according to equality 1), the weights of all arcs (σ, σ') connecting the vertex σ with any other vertex must also be equal to zero. Thus, the application of the elimination operation has led to eliminating arcs whose weight must be equal to zero.

Assume that at a certain stage the vertex σ is eliminated because for a certain $t' \in M(t(\sigma))$ all arcs of the form (σ, σ'), where $\sigma' \in \Omega(t', \sigma)$, have been eliminated. This means that all of these arcs had zero weight. But in this case, in accordance with the same condition 1), the weight of the newly eliminated element σ must be zero. If the necessary condition stated in the theorem is not satisfied, i.e., the grammar $G'(\{\varphi\})$ does not contain an undiminishable subgrammar, then the elimination procedure will continue until in some cell t ∈ T all of the vertices $\sigma \in \bar{\sigma}(t)$ have been eliminated. This means that all of the weights in this cell must be equal to zero. But this contradicts condition 4), which requires that the sum of weights of these vertices be equal to unity. Arriving at such a contradiction, we conclude that the assumption of the absence in grammar $G'(\{\varphi^*\})$ of an undiminishable subgrammar is false. Consequently, if $\{\varphi^*\}$ minimizes the potential function, then $G'(\{\varphi^*\})$ contains an undiminishable subgrammar. Q.E.D.

The grammar $G'(\{\varphi\})$ for optimal $\{\varphi\}$ has a greater significance in view of the following theorem.

THEOREM 3. If $\{\varphi\}$ is a feasible vector in grammar G, then any image that is feasible in grammar $G'(\{\varphi\})$, is feasible in grammar G and yields a maximum of the similarity function.

Proof. It is easily shown that any image feasible in $G'(\{\varphi\})$ is feasible in G as well, since the grammar $G'(\{\varphi\})$ is a subgrammar of G. We shall show that an image that is feasible in the grammar $G'(\{\varphi\})$ is optimal from the viewpoint of the similarity function if we only show that the value of the similarity function on this image is equal to the value of the potential function on the vector $\{\varphi\}$.

Let v* be an image that is feasible in grammar $G'(\{\varphi\})$. This means that there exists a variant z* = (v*, s*) that is also feasible in this grammar.

The equality

$$\sum_{t\in Tv} \max_{\sigma\in\sigma(t)} h(\sigma) = \sum_{t\in Tv} h(z^*(t), t) \tag{7}$$

holds, since $h(z^*(t), t) = \max_{\sigma\in\sigma(t)} h(\sigma)$ for all t ∈ Tv. When these equalities are violated for at least one t the corresponding element z*(t) would not enter into the grammar $G'(\{\varphi\})$ and the variant z* would not be feasible.

By definition we have

$$\sum_{t\in Tv} h(z^*(t), t) = \sum_{t\in Tv} f(v^*(t), x) - \sum_{t\in Tv} \Delta(z^*(t), t). \tag{8}$$

In the course of demonstrating Theorem 1 we elucidated the validity of the equality

$$\sum_{t\in Tv} \Delta(z^*(t), t) + \sum_{t\in Ts} \Delta(z^*(t), t) + \sum_{(t,t')\in N} \delta((z^*(t), t), (z^*(t'), t')) = 0.$$

Each term in the last sum is equal to zero, since otherwise the pair (z*(t), z*(t')) would not enter into grammar $G'(\{\varphi\})$, and thus neither the variant z*. For the same reason every term in the second sum is also equal to zero.

In consequence the equality

$$\sum_{t\in Tv} \Delta(z^*(t), t) = 0,$$

is also true and, when substituted into (8), gives

$$\sum_{t\in Tv} h(z^*(t), t) = \sum_{t\in Tv} f(v^*(t), x). \tag{9}$$

The right side of equality (9) is the value of the similarity function on the image v*, while the left side of (7) is the value of the potential function on the vector $\{\varphi\}$. By virtue of this, the use of (7) and (9) gives $K(\{\varphi\}, x) = F(v^*, x)$. Q.E.D.

THEOREM 4. Let $\{\varphi^*\}$ be a feasible vector in a perfect grammar. Let also $K(\{\varphi^*\}, x)$ be the minimal value of the potential function on the set of feasible vectors. In such a case this value is equal to the maximum value of the similarity function on the set of images, feasible in the grammar G.

Proof. By Theorem 2 the grammar $G'(\{\varphi^*\})$ contains an undiminishable subgrammar $G''(\{\varphi^*\})$. Since the grammar G is perfect, its undiminishable subgrammar $G''(\{\varphi^*\})$ generates at least one feasible image. By Theorem 3 this image yields a maximum of the similarity function. In the course of proving Theorem 3 we also elucidated that the value of the similarity function on this image is equal to the value of the potential function on the vector $\{\varphi^*\}$. Q.E.D.

Theorems 3 and 4 show the way to solve the problem of recognizing real signals, which consists in finding potentials $\{\varphi^*\}$ that minimize the potential function and a feasible image in the grammar $G(\{\varphi^*\})$. If such an image has been found it is optimal in the sense of the similarity function. In the contrary case the algorithm gives the reply "don't know." For perfect grammars the solution of the recognition problem is completed by the minimization of the potential function.

LITERATURE CITED

1. V. N. Vapnik, A. Ya. Lerner, and A. Ya. Chervonenkis, "Pattern recognition training system using generalized portraits," Izv. Akad. Nauk SSSR, Tekh. Kibernetika, No. 1 (1965).
2. V. N. Vapnik and A. Ya. Chervonenkis, Theory of Pattern Recognition [in Russian], Nauka, Moscow (1974).
3. M. A. Aizerman, in: Biological Aspects of Cybernetics [in Russian], Izd. Akad. Nauk SSSR, Moscow (1962).
4. M. A. Aizerman, É. M. Braverman, and L. I. Rozonoér, Potential Functions Methods in the Theory of Learning Machines [in Russian], Nauka, Moscow (1970).
5. M. M. Bongard, in: Biological Aspects of Cybernetics [in Russian], Izd. Akad. Nauk SSSR, Moscow (1962).
6. M. M. Bongard, The Recognition Problem [in Russian], Nauka, Moscow (1967).
7. F. Rosenblatt, Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms, Spartan Books, Washington, D.C. (1962).
8. T. G. Glazkova, K. N. Gurarii, S. I. Danilenko, et al., "Possibilities of applying computers to clinical radiographic differential diagnosis of cancer and benign lesions of the esophagus," Vestn. Radiolog. Rentgenolog., No. 2 (1971).
9. V. S. Morozov et al., "Predicting individual lifetime of electronic instruments using the method of generalized portraits," in: Electronic Engineering, Series I. UHF Electronics, No. 9 (1969).
10. Sh. A. Guberman, M. L. Izvekova, and Ya. I. Khurgin, "The application of pattern recognition methods in the interpretation of geophysical data," in: Self-Learning Automatic Systems [in Russian], Nauka (1966).
11. V. M. Glushkov, Introduction to Cybernetics [in Russian], Naukova Dumka, Kiev (1964).
12. V. A. Kovalevskii (editor), Reading Machines and Pattern Recognition [in Russian], Naukova Dumka, Kiev (1965).
13. V. A. Kovalevskii, "The correlation method of pattern recognition," Zh. Vychisl. Mat. Mat. Fiz., No. 4 (1962).
14. V. P. Romanov and A. A. Savin, "The structurolinguistic method of image recognition," in: Structural Methods of Recognition and Automatic Reading [in Russian], VINITI, Moscow (1970).
15. N. G. Zagoruiko, Recognition Methods and Their Application [in Russian], Sov. Radio, Moscow (1972).
16. A. Martelli and U. Montanari, "Optimal smoothing in picture processing: an application to fingerprints," in: Information Processing 71, North-Holland, Amsterdam (1971).
17. Proceedings of the All-Union Conference on Information Retrieval Systems [in Russian], Vol. 3, VINITI, Moscow (1967).
18. A. Guzman, "Decomposition of a visual scene into three-dimensional bodies," in: Automatic Interpretation and Classification of Images, Academic Press, New York (1969).
19. G. Falk, Computer Interpretation of Imperfect Line Data as a Three-Dimensional Scene, Stanford Artificial Intelligence Project, Memo AIM-132, Computer Science Dept., Stanford University, Stanford (1970).
20. L. A. Zadeh, "Fuzzy sets," Inf. Control, 8, No. 3 (1965).
21. V. A. Kovalevskii, "Optimal decision methods in pattern recognition and the development of reading machines," Doctoral Dissertation, Institute of Cybernetics, Academy of Sciences of the Ukrainian SSR, Kiev (1968).
22. K. S. Fu and T. Huang, "Stochastic grammars and languages," Internat. J. Comput. Inf. Sci., 1, No. 2 (1972).
23. V. A. Kovalevskii, "Optimal algorithm for the recognition of certain sequences," Kibernetika, No. 4 (1967).
24. V. M. Sharypanov et al., Algorithm for the Recognition of Events in Bubble Chambers [in Russian], Izd. Vychisl. Tsentr. Akad. Nauk SSSR, Moscow (1972).
25. P. A. Ota, Mosaic Grammars, University of Pennsylvania (1973).
26. R. Siromoney, "On equal matrix languages," Inf. Control, 14, No. 2 (1969).
27. G. Siromoney, R. Siromoney, and K. Krithivasan, "Abstract families of matrices and picture languages," Comput. Graphics Image Proc., 1, No. 3 (1972).
28. J. L. Pfaltz, "Web grammars and picture description," Comput. Graphics Image Proc., 1, No. 2 (1972).
29. V. A. Kovalevskii, Pattern Recognition: Heuristics or Science? [in Russian], Izd. Inst. Kibernetiki Akad. Nauk UkrSSR, Kiev (1970).
30. M. Minsky and S. Papert, Perceptrons: An Introduction to Computational Geometry, Massachusetts Institute of Technology Press, Cambridge, Massachusetts (1969).
31. A. Martelli, "Edge detection using heuristic search methods," Comput. Graphics Image Proc., 1, No. 2 (1972).
32. H. M. Wagner, Principles of Operations Research with Applications to Managerial Decisions, Prentice-Hall, Englewood Cliffs (1969).