Learnable and Nonlearnable Visual Concepts

HAIM SHVAYTSER

Abstract—**Valiant's theory of the learnable is applied to visual concepts in digital pictures. Several visual concepts that are easily perceived by humans are shown to be learnable from positive examples. These concepts include a certain type of inaccurate copies of line drawings, identifying a subset of objects at specific locations, and pictures of lines in a fixed slope. Several characterizations of visual concepts by templates are shown to be nonlearnable (in the sense of Valiant) from positive-only examples. The importance of representations is demonstrated by showing that even though one can easily learn to identify pictures with at least one of two objects, identifying the objects is sometimes much harder (computationally infeasible).**

Index Terms—**Learnability, machine learning, pattern recognition, visual concepts.**

I. Introduction

OFTEN, learning from examples appears to be easy for humans, but difficult for artificial machines. In this paper we investigate some of the computational aspects that are involved in learning to recognize visual concepts in digital pictures. By a visual concept we mean a collection of objects with "characteristic shape features" that distinguish them from other objects. Examples can be rectangles, triangles, houses, elephants, etc. By learning a visual concept we refer to skills that consist of recognizing whether an object that belongs to a visual concept exists in a picture. Following Valiant [15] we say that a visual concept has been learned if a program for recognizing it has been deduced from examples.

The process of learning a visual concept from examples can be viewed as follows: examples of pictures with a *target concept* are given as input to a learning algorithm. Based on these examples the algorithm produces as output a *rule* for recognizing the target concept, which we call the *learned concept*. The learned concept has to be a "good" approximation of the target concept, but the two are not necessarily identical.

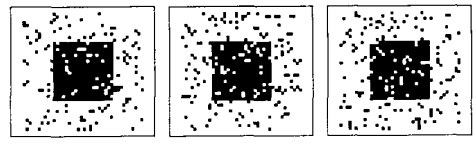
Consider as an example the three binary pictures in Fig. 1. Although the pictures are different, each one appears to have a black square in the center. The target concept here is a central black square. One way of approximately describing this visual concept is by the following rule:

> The average gray level value of the pixels in the central square area is much greater than the average gray level value of the other pixels.

Manuscript received May 23, 1988; revised August 28, 1989. Part of this work was performed while the author was with the Department of Computer Science of Cornell University.

The author is with the David Sarnoff Research Center, Subsidiary of SRI International, Princeton, NJ 08543.

IEEE Log Number 8933766.



(a) (b) (c)

Fig. 1. A central black square.

The output of a learning algorithm (i.e., the learned concept) that gets as input the pictures in Fig. 1 may be: "at least 90% of the pixels in the central square area are black, and at most 10% of the pixels not in the central square area are black." We observe that one can construct examples which follow the above rule but do not look like pictures of a black square, and also examples of pictures with a central black square that do not follow the above rule. However, the rule can be taken as a good description of a central black square in applications where all but a small fraction of pictures with a central black square follow this rule.

The idea that the target concept and the learned concept need not be identical is a part of the complexity based model for learning that was introduced by Valiant in [15], [16]. Valiant's model can be informally described as follows: training examples are drawn randomly according to a fixed but arbitrary probability distribution. (The probability that the examples "occur" in nature.) With respect to this distribution, the learning algorithm produces with "high" probability a learned concept which is a "good" approximation of the target concept. A class of concepts C is learnable in Valiant's model if there is a learning algorithm that for any probability distribution, and for any target concept in C, produces a learned concept with an arbitrarily small probability of mistake, by using a polynomial number of examples and polynominally bounded computational resources. (The polynomial growth is with respect to some "natural" parameters of the concept-class C and the required accuracy.)

To get an intuitive feeling of the difficulty in learning visual concepts that are not polynomially learnable consider an automation that is trained with binary pictures of size 10×10. Assuming that the training requires 1% of all possible examples, and that a new example becomes available every 1 microsecond, the training takes more than 10^{10} million years.

Two important aspects of Valiant's model are that: 1) the learning is required to be asymptotically efficient, and 2) the ability to learn is not limited to specific probability distributions. We maintain that these two aspects are of

great importance in analyzing the ability of machines to learn visual concepts.

First, consider the requirement that recognizing visual concepts can be accomplished by asymptotically efficient programs. Our argument is that since visual concepts are easily recognized by humans in high resolution pictures, it is possible to recognize them by easily computable programs. More specifically:

> Visual concepts in digital pictures can be recognized in time polynomially related to the picture resolution.

The emphasis on asymptotic complexity in analyzing visual perception differs from traditional computational approaches (e.g., [6]). Here, a technique that is inherently exponential is considered inadequate even if it can be efficiently implemented for, say, a 512 × 512 gray level picture. A similar approach was recently taken by Tsotsos [14] to reason about the complexity of visual search.

The other important aspect of Valiant's model is that the efficiency of the learning procedure is required not to be affected by the complexity of the probability distribution from which the examples are obtained. Indeed, usually there is no way of determining this distribution. Still, our ability to learn appears to be independent of this distribution. The key to learning in the distribution free model is the ability to sample examples during the training from their "natural" distribution. Unfortunately, this may not be possible when the training is in a batch style (all training examples are given at once).

As an example, consider the problem of training an automaton to recognize a digitized binary pattern of the printed letter "**A**". Fig. 2 shows several examples of digitized letters. As positive examples one can take pictures such as A1, A2, and A3. Getting "natural" counterexamples for the training is more difficult. Using examples such as B1, B2, and B3 as counterexamples for the training may give absurd results such as identifying the picture X or R in Fig. 2 as "**A**". On the other hand, trying to cover all possibilities such as the pictures X and R as counterexamples may cause many mistakes in classifying "**B**"'s. The difficulty here is that we would like to train the automaton to recognize the letter "**A**", without constraining the allowed counter examples. This is more difficult than merely separating "**A**"'s from, say, "**B**"'s.

The model of learnability that was suggested by Valiant in [15] did not require counterexamples; the more general model that was developed in [16], [1], [5] requires both examples and counterexamples. Computational limitations on learning (in the sense of Valiant) without counterexamples were pointed out in [5], [8], [13]. It was shown that there are simple boolean formulas that cannot be learned from a polynomial number of examples. Many more concept classes are learnable from a polynomial number of both examples and counterexamples [1], [10].

Because of the difficulty to obtain "natural" counterexamples for the training, we focus our analysis on the ability to learn visual concepts *without* counterexamples. However, we assume that there is a way of sampling positive examples from their "natural" distribution, and we require the learning procedure to perform reliably for all such distributions.

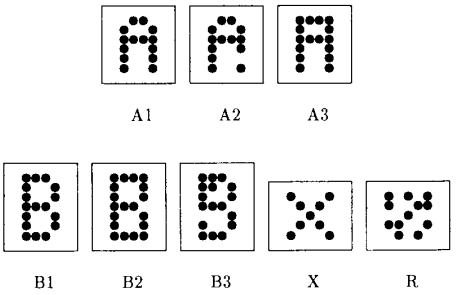


Fig. 2. Digitized letters. A1, A2, A3: the letter A. B1, B2, B3: the letter B. X: the letter X. R: "random" noise.

Learning under these assumptions about the distribution of examples is different from classical pattern classification approaches. Most of the classical techniques for pattern classification depend on *a priori* knowledge of the probability distributions. (See, for example, the Bayesian techniques in [3].) In practice, these distributions are assumed to be normal or uniform, but such assumptions are justified only when the examples are mildly corrupted versions of a single typical picture. Other techniques, such as the perceptron [7] (and maybe other types of neural networks) can be applied without knowledge of the probability distributions, but their training requires both positive and negative (counter) examples drawn from a "natural" distribution. Furthermore, the analysis in [7], [12] of learnable visual concepts does not distinguish between concepts learnable in polynomial time and concepts that are not learnable in polynomial time.

A definition of learnable visual concepts (in the sense of Valiant) is given in Section II. The relevance of learnability results to the design of artificial systems capable of learning visual concepts is discussed in Section III. Several visual concepts that can be learned from positive-only examples are described in Section IV. Section V gives a tool for identifying many characterizations of visual concepts as nonlearnable from positive-only examples.

II. Preliminary Definitions

For simplicity, we consider only the case of binary (black/white) pictures. (Our results can be generalized to the gray level case where each pixel value is determined by a fixed number of bits, independent of the picture resolution.)

A (binary) *digital picture* of size $n \times n$ is an $n \times n$ matrix $v = (v_{ij})$ of 0/1 values. Unless stated otherwise, the term "picture" is used for a binary digital picture. A *visual concept* is a subset of the set of all $n \times n$ pictures.

A picture that belongs to a visual concept is a *positive example* of that concept, and other pictures are *negative examples* (counterexamples) of that concept. When the picture v is a positive example of the concept c we write $v \in c$, and when v is a negative example of c, $v \notin c$.

To investigate the asymptotic complexity of concept learning we consider *families* of concepts that depend on the resolution parameter n. For example, the family of $n \times n$ pictures with a central horizontal line (see Fig. 3) can be expressed as:

$$v_{i,j} = \begin{cases} 1 & i = (n+1)/2 \\ 0 & \text{otherwise.} \end{cases}$$

Throughout the paper we use the convention that a concept is a family of concepts whenever it is defined in terms of arbitrary n variables.

Not surprisingly, the ability to learn a class of concepts (target concepts) may depend on the choice of *representations* that the learning algorithm can use as learned concepts. Let C_n be a class of families of target concepts, and H_n a class of families of representations. We use a definition of learnability that is essentially the same as the ϵ, δ definition that was given in [5], [4], where ϵ is an accuracy parameter and δ a confidence parameter.

Definition 1: C_n is learnable by H_n if there is a learning algorithm such that for all ϵ, $\delta > 0$ and $n \geq 1$, for all target concepts $c \in C_n$, and for all probability distributions D^+, D^- over the positive and negative examples of c respectively:

a) The algorithm gets as input N^+ positive examples and N^- negative examples that are obtained by sampling according to the probability distributions D^+ and D^-, respectively. N^+ and N^- are bounded by a polynomial in n, $1/\epsilon$, $1/\delta$.

b) The algorithm runs in time polynomial in n, $1/\epsilon$, $1/\delta$.

c) With probability of at least $(1 - \delta)$ the output of the algorithm is a learned concept $h \in H_n$ such that for all $n \times n$ pictures v the condition $v \in h$ can be checked in time polynomial in n, $1/\epsilon$, $1/\delta$, and:

$$E^+ = \sum_{\substack{v \in c \\ v \notin h}} D^+(v) < \epsilon. \tag{1a}$$

$$E^- = \sum_{\substack{v \notin c \\ v \in h}} D^-(v) < \epsilon. \tag{1b}$$

Definition 2: C_n is learnable if there exists an H_n such that C_n is learnable by H_n.

It was observed in [15], [8], [5] that when the learning is from positive-only examples, (1b) implies that no mistakes can be permitted in classifying negative examples as being positive. In this case, Definiton 1 is reduced to Definition 3.

Definition 3: C_n is learnable by H_n from positive-only examples if there is a learning algorithm such that for all ϵ, $\delta > 0$, and $n \geq 1$, for all target concepts $c \in C_n$, and for all probability distributions D^+ over the positive examples of c:

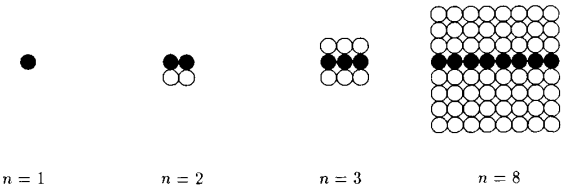


Fig. 3. Several members of the family of pictures with a central black line.

a) The algorithm gets as input N^+ positive examples that are obtained by sampling according to the probability distribution D^+. N^+ is bounded by a polynomial in n, $1/\epsilon$, $1/\delta$.

b) The algorithm runs in time polynomial in n, $1/\epsilon$, $1/\delta$.

c) With probability of at least $(1 - \delta)$ the output of the algorithm is a learned concept $h \in H_n$ such that for all $n \times n$ pictures v the condition $v \in h$ can be checked in time polynomial in n, $1/\epsilon$, $1/\delta$, and:

$$E^+ = \sum_{\substack{v \in c \\ v \notin h}} D^+(v) < \epsilon. \tag{2}$$

d) $v \in h \Rightarrow v \in c$.

Definition 4: C_n is learnable from positive-only examples if there exists H_n such that C_n is learnable by H_n from positive-only examples.

A. Learnability of Boolean Formulas

Many results are known about the learnability (in the sense of Valiant) of boolean formulas. Results that will be used later in the paper are described in this section. (In learning boolean formulas, the concepts are families of boolean formulas and the examples are boolean vectors.)

1) Learnability of **k-CNF**: Let $x_1, \ldots, x_n$ be n boolean variables. A conjunctive normal form (**CNF**) is a conjunction $p_1 \wedge \cdots \wedge p_r$ of clauses, where each clause p_i is a disjunction $q_1 \vee \cdots \vee q_{j_i}$ of literals. A literal is either a variable x or the negation $\bar{x}$ of a variable. A **k-CNF** is a **CNF** expression with clauses that are disjunctions of at most k literals. For example, $(x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee \bar{x}_2 \vee x_3)$ is a 3-**CNF**.

Valiant has shown in [15] that for a constant k (independent of n), **k-CNF** is learnable by **k-CNF**. Valiant's algorithm for learning **k-CNF** with accuracy of $1 - \epsilon$ and confidence of $1 - \delta$ requires $N = (2/\epsilon)(\ln(1/\delta) + n^k)$ randomly chosen positive examples.

Valiant's algorithm:

Let S be the set of all clauses of size at most k.

Let $X_1, \ldots, X_N$ be the randomly chosen examples, where $N = (2/\epsilon)(\ln(1/\delta) + n^k)$.

For each example X_i, $i = 1, \ldots, N$, for each clause $p \in S$ remove p from S if $X_i \notin p$.

Output the conjunction of all the remaining clauses in S.

Example: For $k = 2$, $n = 2$, we have:

$$S = \{x_1, \bar{x}_1, x_2, \bar{x}_2, x_1 \vee x_2, x_1 \vee \bar{x}_2, \bar{x}_1 \vee x_2, \bar{x}_1 \vee \bar{x}_2\}.$$

For the two positive examples $X_1 = (0, 0)$, and $X_2 = (1, 1)$, the algorithm removes the clauses $\{x_1, x_2, x_1 \vee x_2\}$ after the first example, and the clauses $\{\bar{x}_1, \bar{x}_2, \bar{x}_1 \vee \bar{x}_2\}$ after the second example. The algorithm output is the conjunction of the remaining clauses: $(x_1 \vee \bar{x}_2) \wedge (\bar{x}_1 \vee x_2)$.

2) Learnability of **k-term-DNF**: Let $x_1, \ldots, x_n$ be n boolean variables. A disjunctive normal form (**DNF**) is a disjunction $t_1 \vee \cdots \vee t_r$ of terms where each term t_i is a conjunction $q_1 \wedge \cdots \wedge q_{j_i}$ of literals. A literal is either a variable x or the negation $\bar{x}$ of a variable. A **k-term-DNF** is a **DNF** expression with at most k terms. For example, $(x_1 \wedge \bar{x}_2) \vee (\bar{x}_1 \wedge \bar{x}_2 \wedge x_3)$ is a **2-term-DNF**.

It was shown in [9] that **k-term-DNF** can be learned by **k-CNF** from positive-only examples (e.g., by using Valiant's algorithm), but **k-term-DNF** is nonlearnable by **k-term-DNF** *even when both positive and negative examples are available for training*. In fact, even for $k = 2$ the problem is computationally difficult. One cannot learn (in the sense of Valiant) **2-term-DNF** by representations that are also **2-term-DNF** unless the two complexity classes R and NP are the same. See [9] for details.

B. A Nonlearnability Criterion

Let $f_n(x_1, \ldots, x_n)$ be a boolean formula of n variables. Let **PERM**(f_n) be the set of all boolean formulas that can be obtained from f_n by a permutation of variables. For example, if $f(x_1, \ldots, x_4) = (x_1x_2 \vee x_3x_4)$ then:

$$\mathbf{PERM}(f) = \{(x_1x_2 \vee x_3x_4), (x_1x_3 \vee x_2x_4), (x_1x_4 \vee x_2x_3)\}.$$

The following criterion for nonlearnability from positive-only examples has been proved in [13].

The Nonlearnability Lemma: Let f_n be a family of boolean formulas of n variables and C_n be a concept class such that **PERM**$(f_n) \subset C_n$. Let $\mathbf{NEG}_m(n) > 0$ be the number of n coordinate boolean vectors with $m(n)$ "1" coordinates (and $n - m(n)$ "0" coordinates) that are negative examples of f_n. If for any constant α,

$$\lim_{n \to \infty} \frac{n^\alpha \cdot \mathbf{NEG}_m(n)}{\binom{n}{m(n)}} = 0$$

then C_n is nonlearnable from positive-only examples.

III. Teaching Machines How to Learn

The lack of a complete theory of perception limits our ability to design artifical systems capable of recognizing visual concepts. In most cases it is impossible to give an explicit definition (such as an unambiguous equation) of even "simple" concepts such as pictures with a central black square. In this section we discuss the relevance of learning theory to the design of systems capable of recognizing visual concepts.

The standard pattern classification model (e.g., [3]) is usually viewed as composed of three parts: a camera, a feature extractor, and a classifier. The camera produces digital pictures. The feature extractor extracts presumably relevant information from the pictures. The classifier uses this information with a decision rule to classify each picture into one of a small number of categories. Our role in designing systems of this type is to come up with the appropriate features to extract, and the decision rule to be used by the classifier. Unfortunately, this may be an extremely hard task.

The learning approach is to let the system tune itself automatically. Here, instead of a single decision rule, the system can adjust itself by choosing a rule from a whole class of decision rules, based on training examples. In designing systems of this type our role is to decide what features to extract, and what class of decision rules are to be considered. In addition, we have to supply the learning procedure and provide a sufficient number of training examples.

An appropriate choice of features and decision rules should be based on the following:

1) The class of decision rules should be rich enough to potentially enable correct classification.

2) The class of decision rules should be restricted enough to enable efficient learning.

Valiant's learning theory can suggest concrete guidelines for choosing an appropriate class of decision rules and features. The process of determining appropriate decision rules should roughly follow the following steps:

1) Choose a class of decision rules and features that are rich enough to enable correct classification.

2) Determine whether the rules chosen in 1) are learnable from examples. A positive answer should include an appropriate class of representations, a learning algorithm, and the number of examples that the algorithm requires for reliable training.

3) If the class of rules that were chosen in 1) is nonlearnable from examples, consider the following alternatives:

   a) If learning is difficult because of specific representations, try alternative representations as outputs of the learning procedure.

   b) If learning is difficult with positive-only examples, and "natural" counterexamples can be obtained for the training, look for learning algorithms that use both types of examples.

4) If learnability cannot be determined in 3), reexamine the class of decision rules that were chosen in 1) by looking for additional constraints in the problem that may simplify the class of rules.

As was discussed in the introduction, we expect to find simplifying constraints in step 4) whenever the visual concepts are learnable by humans. These ideas will be demonstrated in Section IV.

IV. Learnable Visual Concepts

Using Valiant's algorithm for learning **k-CNF** we show how several families of visual concepts can be learned

from positive-only examples by choosing small clauses of pixels as features, and **k-CNF** expressions as the class of representations (decision rules). We observe that it is not necessary to know in advance which of the concepts is being learned, since the learning algorithm is the same in all cases.

A. Inaccurate Copies

The first case that we consider is the visual concept of a certain type of inaccurate copies, such as pictures of line drawings that are copied with a "shaky" hand. As an example, Fig. 4(b) and (c) are inaccurate copies of the letter "M" in Fig. 4(a).

We proceed to show that under a certain interpretation, this type of inaccurate copies can be learned from positive-only examples by using a feature extractor that produces as features all clauses (with pixels used as binary variables) of size bounded by k^2. Here, k can be viewed as the "shakiness" parameter, corresponding to the amount of inaccuracy that can be tolerated.

The "shaky" hand approximately follows the original pattern. We assume that the mistake in copying never gets too big, and is bounded by a distance of $k/2$ pixels for a constant k independent of the resolution parameter n. Let a $k \times k$ neighborhood of a pixel be the set of pixels in a distance of $k/2$ or less from that pixel. We define the visual concept *k-inaccurate-copies* in the following way:

> Let P be a digital picture of size $n \times n$. A k-inaccurate copy of P is a picture Q of size $n \times n$ such that for each black pixel in P there is at least one black pixel in the corresponding $k \times k$ neighborhood of the pixel in the picture Q, and for each black pixel in Q, there is at least one black pixel in the corresponding $k \times k$ neighborhood in P.

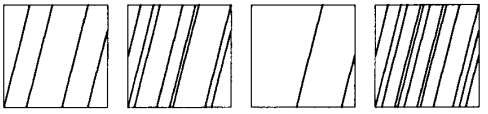
Having at least one black pixel in a $k \times k$ neighborhood can be expressed as a disjunction of k^2 literals. Therefore, the concept of k-inaccurate copies can be expressed as a **CNF** expression with (at most $2n^2$) clauses each of size k^2. Thus, Valiant's algorithm will output (with high confidence) a decision rule that recognizes k-inaccurate copies. To deduce a decision rule of this type that will recognize k-inaccurate copies of a given $n \times n$ picture with accuracy of $1 - \epsilon$ and confidence of $1 - \delta$ Valiant's algorithm requires $(2/\epsilon)(\ln (1/\delta) + n^{2k^2})$ positive-only examples.

B. Straight Lines in a Fixed Slope

We have shown that k-inaccurate copies of a picture car be expressed as a **k-CNF** expression, and are, therefore, learnable by Valiant's algorithm. When the algorithm gets as input positive examples of k-inaccurate copies, it produces (with high confidence) a decision rule that recognizes k-inaccurate copies (with high accuracy). However, if the input pictures are *not* examples of k-inaccurate copies, but of another visual concept that can be represented by **k-CNF**, the algorithm outputs a decision rule for that other concept.

(a) (b) (c)

Fig. 4. Inaccurate copies of the letter "M".



(a) (b) (c) (d)

Fig. 5. Straight lines in a fixed slope.

Consider the pictures of Fig. 5. They are not k-inaccurate copies of any picture, but are examples of straight lines in a fixed slope. Here, a black (white) pixel implies that all pixels on a line with a fixed slope that passes through the pixel are black (white). As in the previous case, we associate boolean values with (binary) pixel values. For a pixel $x_{i,j}$ we have: $x_{i,j} \rightarrow x_{k,l}$ whenever $(l - j)/(k - i) =$ slope. Since $x_{i,j} \rightarrow x_{k,l}$ can also be written as $\bar{x}_{i,j} \vee x_{k,l}$, the concept of straight lines in a fixed slope can be expressed as a 2-CNF expression. Therefore, Valiant's algorithm produces a decision rule for recognizing this concept in $n \times n$ pictures with confidence of $1 - \delta$ and accuracy of $1 - \epsilon$ from $(2/\epsilon)(\ln (1/\delta) + n^4)$ positive examples.

C. k-Object-Disjunctions

Consider the pictures in Fig. 6. Each picture has either a square at the upper left corner, or a triangle at the upper right corner (or both). Similar sequences appear sometimes in intelligence tests, and the subject is required to classify several additional pictures according to whether they follow the same pattern.

We begin with a model of classifying rules for visual concepts of this type, which we call *k-object-disjunctions:*

> $n \times n$ pictures of k-object-disjunctions can be characterized by a classifying rule that can be expressed as a disjunction of k conditions, each of which requires that a particular object appears in a particular position in the picture.

Unfortunately, the above definition is still ambiguous because of the ambiguity in the term "object". Following the guidelines of Section III we choose a very general definition of an object, defining an object as a collection of pixels at particular locations. With this definition of objects, k-object-disjunctions can be modeled as **k-term-DNF** expressions (see Section II-A-2).

Example: In 3×8 pictures with the pixels labeled $x_{1,1}, \ldots, x_{3,8}$ ($x_{1,1}$ being the upper left pixel) having either the black square or the line that are shown in the following

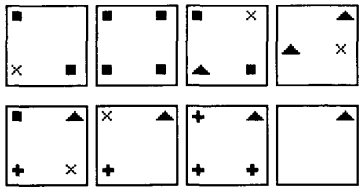
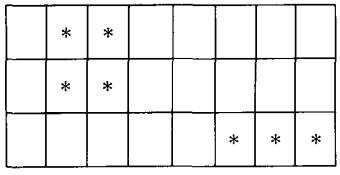


Fig. 6. Two-object-disjunctions.

picture can be expressed by the **2-term-DNF** expression:

$$(x_{1,2} \wedge x_{1,3} \wedge x_{2,2} \wedge x_{2,3}) \vee (x_{3,6} \wedge x_{3,7} \wedge x_{3,8}).$$



The results of [9] (see Section II-A-2), that **k-term-DNF** is nonlearnable by **k-term-DNF** even when both positive and negative (counter) examples are available for the training imply that *one should not attempt to design an artificial system for learning k-object-disjunctions that identifies the k objects*. (Identifying the k objects of a k-object-disjunction is in this case computationally equivalent to determining the **k-term-DNF** expression.) Following Section III we should either consider a refined definition of objects by maybe adding constraints such as connectivity and size, or look for different representations.

In this case, since **k-term-DNF** is learnable by **k-CNF** from positive only examples, Valiant's algorithm can be used to learn k-object-disjunctions without a refinement of our definition of objects. To deduce a **k-CNF** decision rule that will recognize k-object-disjunctions in $n \times n$ picture with accuracy of $1 - \epsilon$ and confidence of $1 - \delta$ Valiant's algorithm requires $(2/\epsilon)(\ln (1/\delta) + n^{2k})$ positive-only examples.

We conclude that Valiant's algorithm (the same algorithm as the one that will produce decision rules for k-inaccurate-copies and for lines in a fixed slope) produces a decision rule that will enable to correctly recognize (with high confidence and accuracy) pictures of k-object-disjunctions without knowing what the objects are!

V. Nonlearnable Visual Concepts

In this section we develop a tool which can be used to show that certain characterizations of visual concepts are too general, and cannot be learned (in the sense of Valiant) from a polynomial number of positive examples regardless of the type or representations that are used by the algorithm.

There are many characterizations of visual concepts in which the classifying rule can be given in terms of a specific picture. Two examples were given in Section IV:

1) The characterization of k-inaccurate-copies of a picture λ is given in terms of the picture λ.

2) A characterization of k-object-disjunctions when $k = 1$ can be given in terms of a picture. Putting $k = 1$ in the definition in Section IV-C we have: "pictures of 1-object disjunctions can be characterized by a classifying rule which requires that a particular object λ appears at a particular position in the picture."

Other examples can be:

- The set of pictures that can be obtained by rotations of a picture λ.
- The set of pictures that can be obtained by a linear transformation of a picture λ.
- The set of pictures with the average gray level the same as the average gray level of a picture λ.
- etc.

(It can be shown that all the examples above are learnable from positive-only examples.) We call the picture λ that appears in these characterizations *a template*.

Definition: Let C_n be a class of families of concepts. We say that C_n can be characterized by templates if

1) Every $c \in C_n$ has a representation of the form $h(\lambda)$, where λ is a template (picture).

2) If $c_1, c_2 \in C_n$, c_1 has the representation $h(\lambda_1)$ and c_2 the representation $h(\lambda_2)$, then

$$c_1 \neq c_2 \Rightarrow \lambda_1 \neq \lambda_2.$$

Thus, if C_n can be characterized by templates then there is a 1–1 (but not necessarily onto) mapping from concepts to templates. We observed that for certain templates it can be that $\lambda_1 \neq \lambda_2$ but $h(\lambda_1) = h(\lambda_2)$, and that there may be templates λ such that $h(\lambda)$ is not a visual concept in C_n.

Determining constraints on the templates λ to guarantee that $h(\lambda)$ is a valid visual concept may be difficult, and sometimes, as was shown in Section IV, unnecessary. Without such constraints the function $h(\lambda)$ is onto. The following lemma can be used to show that sometimes such constraints are necessary.

The Nonlearnability Lemma for Visual Concepts: Let C_n be a class of families of visual concepts such that:

a) C_n can be characterized by a family of templates λ_n.

b) For any template λ of size $n \times n$, $h(\lambda)$ is a characterization of a visual concept in C_n.

Let $Q_\lambda(n, m)$ be the number of $n \times n$ pictures with m black pixels (and $n^2 - m$ white pixels) that are negative examples of the visual concept $h(\lambda)$. If there is a series of numbers $m(n)$ such that:

1) $Q_\lambda(n, m(n)) > 0$.

2) For any α,

$$\lim_{n \to \infty} \frac{n^\alpha \cdot Q_\lambda(n, m(n))}{\binom{n^2}{m(n)}} = 0.$$

Then C_n is nonlearnable from positive-only examples.

Proof: The proof follows from the nonlearnability lemma for boolean formulas of Section II-B. The characterization $h(\lambda)$ corresponds to the boolean function $f(x_1, \ldots, x_n)$, and the pixels of λ correspond to the vari-

ables $x_1, \ldots, x_n$. Condition b) guarantees that the set of characterizations $\{h(\lambda)\}$ is closed under permutation of variables (pixels of λ). □

We say that a template λ of size $n \times n$ matches an $n \times n$ picture P if all the pixels in P at locations that correspond to black pixels in λ are also black. Template matching is a commonly used technique in the analysis of digital pictures [11]. To demonstrate how the nonlearnability lemma for visual concepts can be used to prove nonlearnability of visual concepts we consider two characterizations of visual concepts that are based on template matching:

- *A forbidden template.* Positive examples of this concept *do not* match a template λ.
- *Approximate matching.* Positive examples of this concept match more than a certain percentage of the black pixels in a template λ.

We show that without additional constraints on the templates these visual concepts are nonlearnable (in the sense of Valiant) from positive-only examples.

A. Nonlearnability of Forbidden Templates

To apply the nonlearnability lemma[1] let $d(n)$ be the number of black pixels in the forbidden template λ. Take the value of $m(n)$ in the lemma as $m(n) = d(n)$. Since only one picture with d black pixels matches a given template λ with d black pixels, $Q_\lambda(n, m) = 1$, and forbidden templates cannot be learned from positive examples whenever:

$$\text{for any } \alpha, \lim_{n\to\infty} \frac{n^\alpha}{\binom{n^2}{d}} = 0.$$

The above condition holds whenever d and $n^2 - d$ are unbounded. Thus, forbidden templates cannot be learned if their size is not a constant, independent of the picture resolution. However, if their size is independent of the picture resolution they can only match infinitesimally small objects in high resolution pictures or the entire picture.

B. Nonlearnability of Approximate Template Matching

To apply the nonlearnability lemma for the approximate matching of a template λ with d black pixels, let us denote by $z \cdot d$ the threshold value of the approximate match, i.e., a picture is an approximate match of the template λ if it matches *more* than $z \cdot d$ of the template black pixels $(0 < z < 1)$.

Now take $m(n) = n^2 - (1 - z)d$, i.e., the set of examples consists of pictures with $(1 - z)d$ white pixels and $n^2 - (1 - z)d$ black pixels. Such pictures are *not* an approximate match of the template only if *all* the white pixels match black pixels of the template. Therefore,

$$Q(n, m) = \binom{d}{(1 - z)d} = \binom{d}{zd},$$

and the nonlearnability condition of the lemma is:

$$\lim_{n\to\infty} \frac{n^\alpha \binom{d}{zd}}{\binom{n^2}{(1 - z)d}} = 0. \tag{3}$$

We will show that it holds whenever $d = z' \cdot n^2$, for all $0 < z' < 1$. (The template size is a fixed percentage of the picture size.) In this case, (3) is:

$$\lim_{n\to\infty} \frac{n^\alpha \binom{z'n^2}{zz'n^2}}{\binom{n^2}{(1 - z)z'n^2}} = 0. \tag{4}$$

Both numerator and denominator grow exponentially as a function of n. The exponential component of a binomial coefficient of this type is given by (see [2]):

$$\binom{N}{c \cdot N} \approx \left(\frac{\left(\frac{1}{c} - 1\right)^c}{1 - c}\right)^N \quad \text{for a constant } 0 < c < 1.$$

Substituting in (4), we conclude that the nonlearnability condition holds whenever

$$\left(\frac{(1/z - 1)^z}{1 - z}\right)^{z'} < \frac{\left(\frac{1}{(1 - z)z'} - 1\right)^{(1-z)z'}}{1 - (1 - z)z'}$$

and this can be shown to hold for all constants $0 < z < 1, 0 < z' < 1$.

What about templates of smaller size? When $d(n)$ is a function of n that increases no faster than a logarithm, the proof is even simpler. Let $d(n)$ be monotone increasing, with $d(n) < \beta \log n$. We have: $\binom{d}{zd} < 2^d < n^\beta$ for a constant β. Therefore, the numerator of (3) grows only polynomially fast, while the denominator grows at least at super-polynomial rate when $d(n)$ is unbounded.

VI. Concluding Remarks

In this paper we have considered the process of learning to identify visual concepts from examples, without being given an explicit description of these concepts. Valiant's complexity based model was used to investigate the learnability of visual concepts in digital pictures by considering the learning complexity as a function of the picture resolution. (A class of visual concepts is learnable if the complexity of learning concepts from this class grows polynomially as a function of the resolution.) This approach is motivated by psychological evidence which suggests that humans have no difficulty in learning to recognize visual concepts in high resolution pictures.

An important aspect of Valiant's model is that the learner learns to recognize very accurately examples that he may encounter with high probability. The behavior of

[1]The nonlearnability for $d = n/2$ can also be proved by using Theorem 15 in [5].

the recognition algorithm for examples with very low probability is irrelevant. However, the efficiency of the learning procedure is required not to be limited by the complexity of the probability distributions of the examples, which may be arbitrarily complex.

There are many difficulties in any attempt to reason about human visual learning by applying complexity based arguments. On one hand it does not seem likely that humans can learn all that can be learned in polynomial time. It may be that many classes of visual concepts that require polynomial time to learn are too complex for humans. On the other hand, nonlearnable visual concepts may become learnable if something is known (or assumed) about the probability distribution of the examples. The major contribution of the complexity based analysis of learning is in suggesting concrete guidelines for designing *artificial systems* capable of learning visual concepts.

References

[1] A. Blumer, A. Ehrenfeucht, D. Haussler, and M. Warmuth, "Learnability and the Vapnik-Chervonenkis dimension," *J. ACM*, vol. 36, pp. 929-965, Oct. 1989.

[2] B. Bollobas, *Extremal Graph Theory*. New York: Academic, 1978.

[3] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[4] D. Haussler, M. Kearns, N. Littlestone, and M. Warmuth, "Equivalence of models for polynomial learnability," in *Proc. 1st Workshop Computational Learning Theory*. Morgan Kaufmann, 1988, pp. 42-55.

[5] M. Kearns, M. Li, L. Pitt, and L. G. Valiant, "On the learnability of boolean formulae," in *Proc. 19th Annu. ACM Symp. Theory of Computing*, May 1987, pp. 285-295.

[6] D. Marr, *Vision*. San Francisco, CA: Freeman, 1982.

[7] M. Minsky and S. Papert, *Perceptrons: An Introduction to Computational Geometry*. Cambridge, MA: MIT Press, 1969.

[8] B. K. Natarajan, "On learning boolean functions," in *Proc. 19th Annu. ACM Symp. Theory of Computing*, May 1987, pp. 296-304.

[9] L. Pitt and L. G. Valiant, "Computational limitations on learning from examples," *J. ACM*, vol. 35, no. 4, pp. 965-984, Oct. 1988.

[10] R. Rivest, "Learning decision-lists," *Machine Learning*, vol. 2, no. 3, pp. 229-246, 1987.

[11] A. Rosenfeld and A. C. Kak, *Digital Picture Processing*, vol. 2, 2nd ed. New York: Academic, 1982.

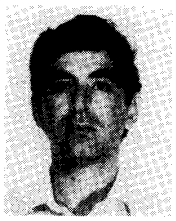
[12] D. E. Rumelhart and J. L. McClelland, *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1986.

[13] H. Shvaytser, "A necessary condition for learning from positive examples," *Machine Learning*, vol. 5, no. 1, pp. 101-103, 1990.

[14] J. K. Tsotsos, "Analyzing vision at the complexity level," *Behavioral Brain Sci.*, vol. 12, no. 3, 1990.

[15] L. G. Valiant, "A theory of the learnable," *Commun. ACM*, vol. 27, no. 11, pp. 1134-1142, 1984.

[16] L. G. Valiant, "Learning disjunctions of conjunctions," in *Proc. 9th IJCAI*, Aug. 1985, pp. 550-556.



Haim Shvaytser received the B.Sc. degree from Tel Aviv University, Tel Aviv, Israel, in 1982, and the Ph.D. degree from the Hebrew University, Jerusalem, Israel, in 1986.

He was a postdoctorate at the University of Texas at Austin, Columbia University, and Cornell University. He is currently a member of the Technical Staff at David Sarnoff Research center in Princeton, NJ. His research interests include computer vision, computational learning, neural networks, and artificial intelligence.