Pattern Recognition Vol. 10, pp. 189-204.
Pergamon Press Ltd. 1978. Printed in Great Britain.

0031-3203/78/0601-0189 $02.00/0

THE USE OF CONTEXT IN PATTERN RECOGNITION*†

GODFRIED T. TOUSSAINT
School of Computer Science, McGill University, Montreal, Quebec, Canada

(*Received* 31 *October* 1977; *received for publication* 3 *January* 1978)

Abstract – The importance of contextual information, at various different levels, for the satisfactory solution of pattern recognition problems is illustrated by examples. A tutorial survey of techniques for using contextual information in pattern recognition is presented. Emphasis is placed on the problems of image classification and text recognition, where the text is in the form of machine and handprinted characters, cursive script, and speech. The related problems of scene analysis, natural language understanding, and error-correcting compilers are only lightly touched upon.

Character recognition Speech recognition Pattern recognition Text processing Error correction Spelling correction Image classification Scene analysis Natural language understanding Context Artificial intelligence

1. INTRODUCTION

The notion and importance of context is known to all of us. It is fundamental to many, if not all, spheres of human endeavour. Artists have courted it for thousands of years and scientists have fought it by manipulating experimental variables in order to observe phenomena "out of context". Physicists have had a great deal of success, in contrast to scientists working with living systems, because physical scientists have fewer variables to deal with and context is of less importance. The new field of pattern recognition is different from the physical sciences in that we are trying to mechanize, or externalize into a physical symbol system, a fundamental ability of living organisms. One could guess from this observation that trying to solve the pattern recognition problem, by emulating the methodology of the physical sciences in trying to obtain context-independent solutions to problems, would be a difficult task. The past 20 years experience testifies that this is, in fact, the case. Furthermore, the new realization that many problems cannot be satisfactorily solved by ignoring context is resulting in an extensive effort to find methods of incorporating contextual information at one level or another. The importance of context is realized and documented in such diverse fields as art,[6,7] philosophy of mathematics,[61] psychology,[40,55,72,104,109,120] anthropology,[79] artificial intelligence,[12,52,78,123] statistics,[42,181] and, of course, pattern recognition.[176,184]

Basically, the effect of context is that some entity Z can have certain properties, when Z is viewed in isolation, which change when Z is viewed in some context. Alternately, an entity Z is seen as one thing in context A and another in context B. This effect can occur at many different levels including perceptual, cognitive, and "objective" mathematical levels. For example, consider the two horizontal lines in Fig. 1(a); they appear to be of the same length. But now add some context in the forms of arrows at the ends so as to obtain Fig. 1(b). In Fig. 1(b) the lower of the two lines now appears to be longer than the upper one. This is the well-known Muller–Lyer illusion.[49,148] As a second example, consider the three acute angles shown in Fig. 2(a); besides these angles nothing else is visible. But now add some context in such a way as to obtain Fig. 2(b). In Fig. 2(b), we see a "whiter than white" triangle made up of lines that are "actually not there." These are known as subjective contours in some circles and hallucinations in others. For a third example, consider the task of reading a word, say, YELLOW. If contextual information is brought in in the form of the colour of ink used in printing the word YELLOW then this information affects the performance of the task. This is known as the Stroop phenomenon.[55] Illusions such as the Muller–Lyer illusion, or subjective contours, and effects such as the Stroop phenomenon, are the results of the effects of context at the perceptual level, particularly, at the visual perceptual level. However, illusions and confusions manifest themselves also in other modalities of the perceptual level,[192] such as the auditory modality in the case of speech recognition. In speech recognition verbal context can make us perceive phonemes that, once again, are "really not there". From the above examples of illusions it may appear that context is a

† This research was supported by the National Research Council of Canada under grant number NRC-A9293.

"villain", but it actually has its "good" side too, as in phonemic restorations when the context is clear but the stimulus is absent. In the words of Warren and Warren[192] "It is curious that in studying illusions and confusions we encounter mechanisms that ensure accurate perception and the appropriate interpretation of ambiguities." The use of semantics to disambiguate a sentence such as "the box is in the pen" is an example of the use of context at the cognitive level. Finally, consider the case of ordering measurements, in statistical hypothesis testing, either individually (out of context) or in pairs (in a context). An amazing result is that, even when the measurements are independent in a statistical sense, the best two measurements are not the two best.[181,42] This illustrates the importance of context at the mathematical level. With respect to the Muller–Lyer illusion, it is interesting to note that the magnitude of the illusion varies with culture.[49] Hence, this phenomenon illustrates an effect of context at the cultural level.

The examples of context at the perceptual level discussed above show how context can help us in perceiving or recognizing something that is "actually not there". On the other hand, sometimes we *cannot* recognize something that *is* "actually there" unless we are given the context. This point is dramatically illustrated in Fig. 3 which is the author's modification of a similar drawing which appeared in the book by Adams.[3] Most people cannot recognize the head of a cow in Fig. 3(a), no matter how long they look at it. However, once they have been shown the surrounding context in Fig. 3(b), the face of the cow is readily perceived in Fig. 3(a). This illustrates that some problems cannot be solved through an ever increasing depth of analysis but, rather, must be solved by widening the context in which the problem is viewed.

Fig. 1(a).

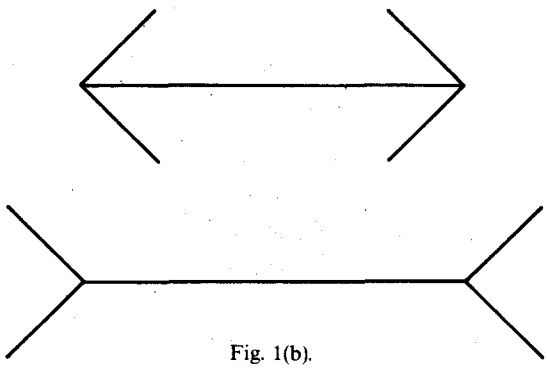


Fig. 1(b).

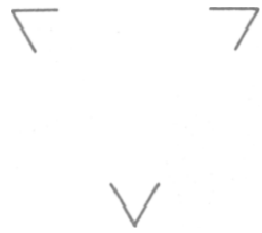


Fig. 2(a).

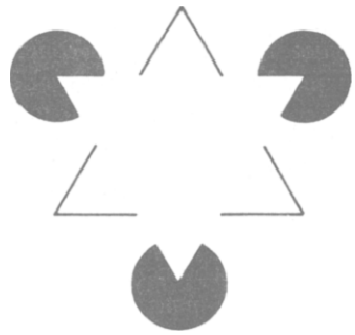


Fig. 2(b).

Fig. 3(a).

The perceptual and cognitive aspects of context discussed above should not be viewed as two distinct categories. Rather they are more like a continuum. A similar continuum can be observed in the field of automatic pattern recognition where the analogous task of perception is image classification and the analogous task of cognition includes scene analysis and natural language understanding. When the latter two are treated mainly from the semantic point of view, they are usually considered as part of artificial intelligence. It is important to realize also that the context, whether perceptual or cognitive, lies not exclusively with the data. Context, in the form of expectations, lies also within the perceiver. Systems that process patterns by analysing the data or input information with ever-increasing levels of sophistication are called *data-driven* or *bottom-up* systems.

Fig. 3(b).

Those that start from overall expectations and work down are called *conceptually-driven* or *top-down* systems. It appears that for solving difficult problems efficiently context may have to be used with both *bottom-up* and *top-down* processing taking place simultaneously. This is particularly important for scene analysis and natural language understanding but also, as we shall see, for text recognition. These concepts are more fully discussed by Norman and Bobrow,[127] and Palmer.[132] A contextual approach to memory can be found in Jenkins,[98] and Sachs.[143] For an introduction to the philosophical position of contextualism the reader is referred to Pepper.[133]

The above examples of how humans appear to use context at the perceptual and cognitive levels suggests that in automated pattern recognition we should be able to use context in order to tackle such problems as (1) disambiguation, (2) error-correction, and (3) filling in gaps. Filling in the gaps usually arises either because information is missing or because information is partially or completely destroyed by noise. The following two sections will review how the above problems are being tackled in image classification and text recognition. No attempt at being exhaustive with respect to such fields as scene analysis, natural language understanding and error-correcting compilers is made, as these fields are tangential, although relevant, to pattern recognition and space is limited.

2. CONTEXT IN IMAGE CLASSIFICATION

Consider an image, represented as an $n \times m$ matrix of black (one) and white (zero) cells, which is to be classified into one of two groups or classes. A typical value of n or m might be 30 and a typical image may be that of a letter, number, a blood cell or a chromosome although larger and more complex images such as that shown in Fig. 4 are certainly possible. An implicit top-down or contextual approach is the old idea of template matching. Keep a set of $n \times m$ templates of all

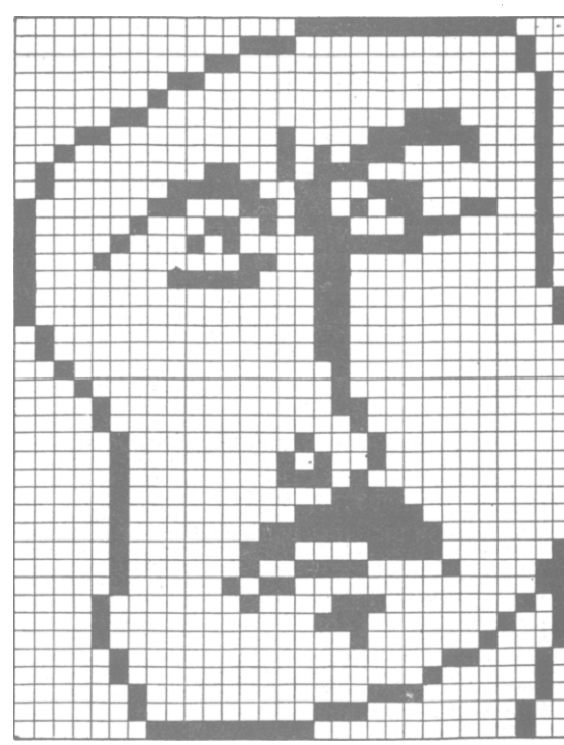


Fig. 4.

the patterns that can occur along with their labels as to what class they belong to. Compare the image as a whole to each template and find the one that matches. This approach is also called *table look-up* or *dictionary look-up*. The trouble with these approaches is the combinatorial explosion in storage and computation due to the fact that $2^{n \times m}$ images are possible. Furthermore, if some images can belong to both classes, probabilities must be estimated and to estimate the distribution of that many states could require an astronomical amount of data. The approach that is therefore chosen is more like a bottom-up approach in which the cells are considered independently and context is incorporated explicitly in the form of dependencies between the image cells. We now have to estimate only $n \times m$ probabilities and the dependence parameters for each class. Alternately, this can be viewed as approximating the original probability distribution of order $n \times m$ by a function of lower-order marginals. Hence, we see that in image classification using contextual information manifests itself as incorporating dependencies between pixels through one method or another. For the case of binary or discrete valued pixels most approaches are based on probability distribution approximations. One way of approximating a high-order distribution with low-order marginals is to assume that the dependencies exist in the form of a low-order Markov chain. In order to handle two dimensional images the chain can be either mapped onto a Hilbert space-filling curve or it can be generalized to a Markov mesh as was done by Abend, Harley and Kanal.[1] A Hilbert space-filling curve, illustrated in Fig. 5, specifies a path connecting each cell or pixel of the image. Although it does have the effect of converting the linear dependence of a Markov chain into a spatial dependence with some nice properties, it suffers from the fact that it may neglect important dependencies between pixels far away from each other in the image. A better method of approximating a distribution is through the use of dependence trees.[35] A dependence tree is illustrated in Fig. 6, where x_κ stands for a pixel. The tree indicates that x_2 is dependent on x_1, pixels x_3, x_4, and x_5, are dependent on x_2, etc.

A basic principle then appears to be that in order to use *statistical contextual information* in an efficient manner it is necessary to use suitable approximations to otherwise unmanageable probability distributions. This will be a recurring theme in scene analysis and text recognition.

The method of classifying images by considering the $n \times m$ cells as basic units will only work for simple problems in which there occurs little variability among patterns. In more difficult problems it is more common to make k measurements (observe k features) on a pattern and consider these as the basic units. Using context then implies taking account of dependencies among features. If the features are discrete, such as *present* or *absent* the distribution approximation approaches discussed above can be taken. If the features

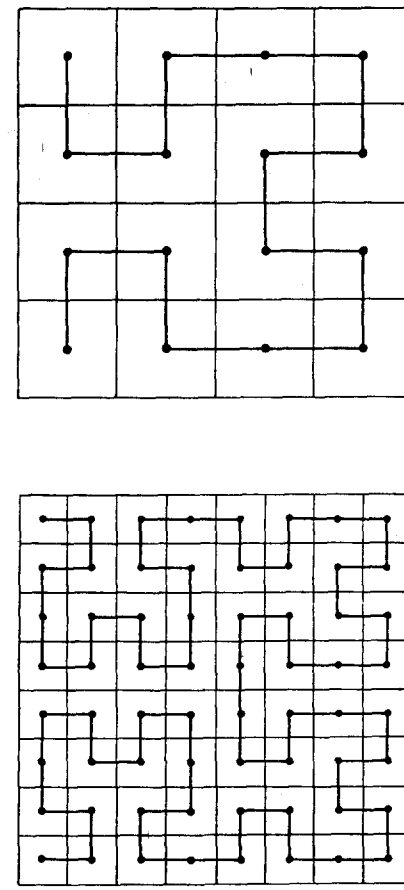


Fig. 5.

are continuous, dependence is often introduced by incorporating a covariance matrix structure. Both of these methods are statistical approaches to using contextual dependencies. Sometimes the features are not arbitrary measurements but are strongly related to the structure of the pattern. When this is true the features are referred to as *primitives*, as in the Chinese characters illustrated in Fig. 7. In Fig. 7, the primitives are clearly related to each other in a syntactic fashion with some primitives being "left-of", "on-top-of", and "inside-of", other primitives. It is then often more convenient to incorporate contextual constraints using *syntactical rules* rather than statistical methods. These approaches to using context fall under the syntactic methods in pattern recognition and can be found in Fu[69] and in Fung and Fu.[70] A closely related problem is the one of designing error-correcting compilers. Various methods for incorporating contextual constraints in error-correcting compilers can be found in Heinselman[68] and Thompson.[172] When patterns are more complicated than, say, Chinese characters (such as office scenes) the problem is referred to as *scene analysis* and syntactic methods have been much preferred (Duda and Hart,[54] Guzman[78] and Rozenfeld[141]), although, recently, probabilistic methods are receiving renewed attention (Zucker[196]).

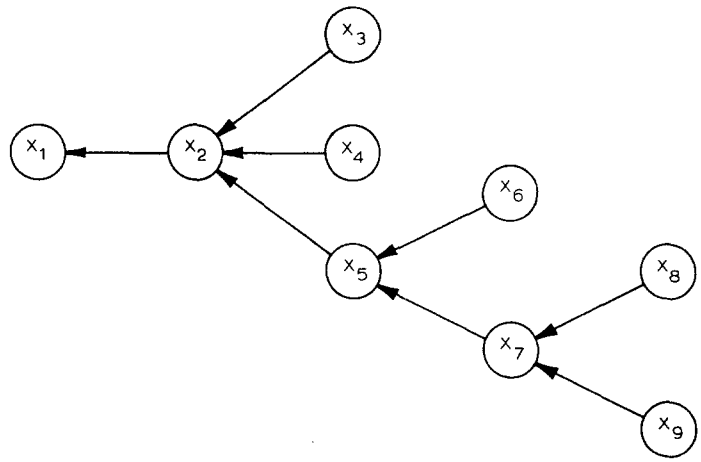


Fig. 6.

3. CONTEXT IN TEXT RECOGNITION

Introduction

It is true that text is made up of letters and spaces (ignoring punctuation marks) and that, if letters and spaces were correctly classified, so would the text. However, errors invariably occur in practice and, hence, a machine which recognizes text by recognizing individual characters will make errors on the text. Humans, on the other hand, have the capacity of correctly recognizing mutilated text[119] although they make errors on characters viewed out of context.[124,178] Psychological studies have shown that single letters can be perceived better when they are parts of words.[193] In fact, evidence seems to show that humans do not recognize the individual letters and construct a word but, rather, they recognize the entire word based on information of various kinds (word features) and, thence, know what the individual letters must be.[72] In other words, letter recognition in the context of words appears to be primarily *top-down* rather than *bottom-up*. Hence, one way for a machine to use context in recognizing letters is to actually recognize words. In fact, this approach is usually preferred in recognition of speech and cursive script in order to bypass the difficult problem of segmentation.

Fig. 7.

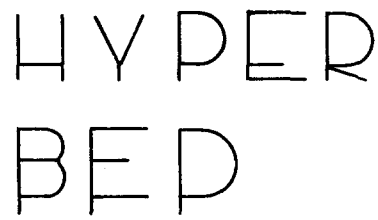


Fig. 8.

A question that comes to mind at this point is: what kind of information is contained in a word that a machine might be able to make use of? It is useful to distinguish between six classes of *word features*: graphological, phonological, statistical, syntactic, semantic, and pragmatic.

Graphological features are those that characterize the shape of a word or the shape of a letter-cluster within a word. An example of a gestalt graphological word feature is the height-to-width ratio of the word. Graphological contextual information can also occur in the form of printing style as was nicely illustrated by Blesser *et al.*[18,44,153] The latter type of graphological feature is illustrated in Fig. 8 with the words HYPER and BED printed in different styles such that the "P" in HYPER and the "D" in BED are the same "character" having the same shape. In other words the character is ambiguous and the ambiguity is resolved by the printing style.

Phonological features are present because humans have an idea of what sounds to expect in certain situations, as in the acoustic similarity of rhyme.

Statistical information is present in words to the extent that the more frequently occurring words are more familiar to us and are more easily recognized.[150]

Syntactic features of words consist of markers for parts of speech.

Semantic features are markers indicating the meaning of words, i.e. taxonomic categories.

Pragmatic features consist of information about how the user uses a word.

In the case of cursive script some of the pragmatic features are intimately related to some of the graphological features.

Contextual information in words can also be expressed in terms of statistical or syntactic rules concerning the relations among letters in a word. The frequencies of occurrence of letters, letter pairs, and letter triplets, constitute examples of statistical rules.

Kashyap and Mittal use many syntactic rules in a speech recognition system. Typical examples of syntactic rules which they used are:

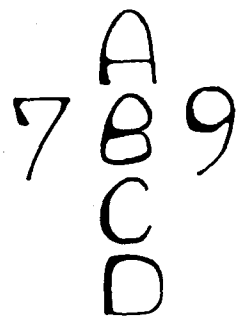
(a) There cannot be two or more consecutive vowels.

(b) There cannot be three or more consecutive consonants.

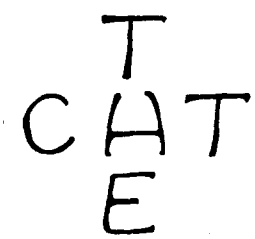
(c) Consonants and only the vowels /*a*/ and /*e*/ can occur at the beginning of a word.

(d) No word can end with a vowel or with the consonant /*h*/.

Knowledge of taxonomic categories, syntactic rules such as those discussed above, and semantic information can all help in error correction and the resolution of ambiguities. Some further examples are illustrated in Figs 9–11. In Fig. 9(a), knowledge of the taxonomic categories "numbers" and "letters" disambiguates the "B-8" character. In Fig. 9(b), the character "H-A" is disambiguated through syntactic rules of the English language – the strings TAE and CHT are not legal. Figure 10 illustrates error-correction through syntactic contextual constraints; the letter "Q" is always followed by the letter "U", and, therefore, the "V" must have been an error. Finally, the sentence in Fig. 11 illustrates the disambiguation of a word by semantic information. If the sentence occurs in the setting of a picnic it probably ends in "CUP". If it occurs in the setting of an office with a desk full of paper, it probably ends in "CLIP".



(a)



(b)

Fig. 9.

Fig. 10.

Fig. 11.

It is intuitively clear that it is easier for a computer to use graphological, statistical, and syntactic information than the other types. In the past, efforts at using context in text recognition have been directed mainly at the problem of using statistical and syntactic information. The approaches that have been taken can be categorized into three groups: dictionary look-up methods, probability distribution approximation techniques, and hybrid methods. All these approaches can be subsumed under the general statistical framework of compound decision theory to which the next section is devoted.

Compound decision theory

Let us assume that we have a text consisting of N characters. Each character yields a feature vector X_i, $i = 1, 2, \ldots, N$. Let $P(\mathbf{X} \mid \lambda = \theta)$ denote the probability of the vector sequence $\mathbf{X} = X_1, X_2, \ldots, X_N$ conditioned on the sequence of identities $\lambda = \lambda_1, \lambda_2, \ldots, \lambda_N$ taking on the values $\theta = \theta_1, \theta_2, \ldots, \theta_N$, where X_k is the feature vector for the k-th character, and where θ_k takes on M values (number of pattern classes) for $k = 1, 2, \ldots, N$. Also, let $P(\lambda = \theta)$ be the *a priori* probability of the sequence λ taking on the values θ. In other words $P(\lambda = \theta)$ is the *a priori* probability distribution of all sequences of N characters. The probability of correctly classifying the text is maximized by choosing that sequence of characters that has a maximum *a posteriori* probability given by $P(\lambda = \theta \mid \mathbf{X})$. From Bayes' rule we obtain

$$P(\lambda = \theta \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid \lambda = \theta) P(\lambda = \theta)}{P(X)}. \tag{1}$$

Since $P(\mathbf{X})$ is independent of the sequence θ we need only maximize the discriminant function

$$g_\theta(\mathbf{X}) = P(\mathbf{X} \mid \lambda = \theta) P(\lambda = \theta). \tag{2}$$

These probabilities would require an astronomical amount of storage in practice and, hence, simplifying assumptions must be made. One assumption which is

reasonable is to let N be the size of a word. Then $P(\lambda = \theta)$ is the frequency of occurrence of words. To simplify $P(\mathbf{X} \mid \lambda = \theta)$ we can assume conditional independence among the feature vectors $X_1, X_2, \ldots, X_N$. This assumption states that, given the true identity of a character, the shape of the character, which generates a given feature vector, is independent of the shapes and true identities of neighbouring characters and is, therefore, dependent only on the true identity of the character in question. While this assumption is not valid for cursive script, as is illustrated in Fig. 12 where the shape of the letter "O" depends on whether the previous letter is an "O" or a "D", its invocation is necessary in order to reduce the problem down to manageable size. Under this assumption, and taking logarithms, (2) reduces to

$$g_\theta(\mathbf{X}) = \sum_{i=1}^{N} \log P(X_i \mid \lambda_i = \theta_i) + \log P(\lambda_1 = \theta_1, \ldots, \lambda_N = \theta_N). \quad (3)$$

For further aspects of compound decision theory the reader is referred to the following works.[2,89,90,93,15]

Dictionary look-up methods

Dictionary look-up methods are among the oldest methods of using contextual information and are quite popular in limited-vocabulary word-recognition systems, particularly, when the words come in the form of speech or cursive script. One of the earliest applications of this method to text recognition was by Bledsoe and Browning.[17] The method is conceptually simple and illustrated by example in Fig. 13. Let us say the machine is fed the word HAT. The classifier computes discriminant functions for each letter in the word, as illustrated in Fig. 13, where $X_i, i = 1, 2, 3$ is the feature vector of the ith letter in the word, and where $g_\theta(X_i)$ is the confidence, in some sense, that given the feature vector X_i, θ is the true class, where $\theta \in \{A, B, \ldots, Z\}$. Assume that there are n words of length three in the dictionary. The contextual decoder "looks" at all words of length three in the dictionary and computes a score $S_i(\cdot), i = 1, 2, \ldots, n$ as in Fig. 13. The word chosen is the one having the highest score. Methods such as this one work well in the sense that they obtain a high error correction rate. This is because the procedure guarantees that only valid strings of characters (words) are put out. Comparing the method to (3) we see that if the discriminant functions are the logarithms of the likelihoods, i.e. if $g_{\theta_i}(X) = \log P(X_i \mid \lambda_i = \theta_i)$ then the Bledsoe–Browning method makes a maximum likelihood decision on a word where θ is constrained to take on values that exist in the dictionary. This suggests an immediate improvement to the Bledsoe–Browning dictionary look-up method. Extend in (3) the scores as follows:

$$S_1(ARE) = g_A(X_1) + g_R(X_2) + g_E(X_3) + f[P(ARE)],$$

where f is some convenient function such as the logarithm, and $P(ARE) = P(\lambda_1 = A, \lambda_2 = R, \lambda_3 = E)$. In error correction or disambiguation this method will obviously favour the more frequently occurring words. This method is closer to the optimal Bayes approach but at the added cost of having to estimate and store the word frequencies. It should be noted, of course, that the probability distributions of order n, where n is the number of letters in a word, could be approximated by functions of lower-order marginals through the methods mentioned in the second section on image classification, thus reducing, perhaps substantially, the storage required for the word-frequency distribution.

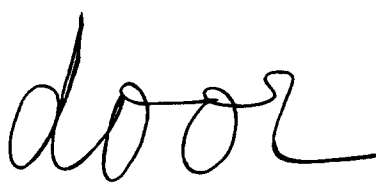


Fig. 12.

The above methods can also be used when no explicit information about the measurements is available. In other words, the input to the contextual decoder can consist of the decisions made by a regular classifier that does not use context. In fact, any text can serve as the input and hence these techniques are often referred to as *degarbling* or *automatic spelling correction*. Spelling correction systems are usually concerned with three types of errors: *insertion*, *deletion*, and *substitution*. *Insertion* errors occur when there is an additional letter in the word, as in CONNTEXT. Deletion errors occur when there is a missing letter in the word, as in CONTXT. Substitution errors, by far the most frequently occurring in text recognition and the topic of this paper, occur when the word containing errors has the same number of letters as the correct word. The substitution errors may be caused by individual letter errors, transposition of two letters (a frequent typing error), deletions and insertions occurring simultaneously, or, of course, any suitable combination of these. As an example of a simple procedure for automatic spelling correction consider the method of substitution-error correction used by Vossler and Branston. In this approach the information contained in the discriminants of alternatives is summarized in a *confusion matrix* obtained through a trial run on some text, under supervision. The contextual decoder then makes use of the dictionary, the *a priori* probabilities of words, and the confusion matrix. The confusion matrix is made up of probabilities of the form $P_c(\lambda = i \mid \lambda = j)$ denoting the probability that a character belonging to class i is classified into class j, $i, j = 1, \ldots, 26$.

Assume that the input to the contextual decoder is the word TAE. In order to perform error correction we must determine which 3-letter word in the dictionary is most likely to have caused TAE, and choose that word, i.e. we must maximize the *a posteriori* probability of the classes θ_1, θ_2, θ_3 given the evidence TAE. In other words, the discriminant function becomes

DICTIONARY LOOK-UP ALGORITHM

INPUT PATTERNS **H** **A** **T**

FEATURE VECTORS X_1 X_2 X_3

DISCRIMINANT FUNCTION VALUES

$$\begin{matrix} g_A(X_1) & g_A(X_2) & g_A(X_3) \\ g_B(X_1) & g_B(X_2) & g_B(X_3) \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ g_Z(X_1) & g_Z(X_2) & g_Z(X_3) \end{matrix}$$

STEP 1: Compute scores for words in DICTIONARY.

$$S_1(ARE) = g_A(X_1) + g_R(X_2) + g_E(X_3)$$
$$S_2(HAT) = g_H(X_1) + g_A(X_2) + g_T(X_3)$$
$$\vdots$$
$$S_n(VAN) = g_V(X_1) + g_A(X_2) + g_N(X_3)$$

STEP 2: Choose the word that gives the highest SCORE value.

Fig. 13.

$g_\theta(TAE) = P(\lambda_1 = \theta_1, \lambda_2 = \theta_2, \lambda_3 = \theta_3 \mid TAE)$, where θ_i, $i = 1,2,3$ takes on those values such that any triplet $[\theta_i\theta_j\theta_k]$ is a word in the dictionary. Using Bayes' rule as before, taking logarithms, and assuming conditional independence yields

$$\begin{aligned} g_\theta(TAE) = {} & \log P(T \mid \lambda_1 = \theta_1) + \log P(A \mid \lambda_2 = \theta_2) \\ & + \log P(E \mid \lambda_3 = \theta_3) \\ & + \log P(\lambda_1 = T, \lambda_2 = A, \lambda_3 = E), \end{aligned} \quad (4)$$

which is similar to (3) where the likelihoods are now confusion probabilities. The expression (4) must be computed for all words of length three in the dictionary and that word with the highest score is chosen as the word intended.

It is clear that in all the above methods the computation and storage is a function of the number of words in the dictionary. Hence, computation and storage become prohibitive when dictionaries are of a conveniently large size for many applications. These shortcomings have led to the development of the Markov methods which fall under the more general class of approaches which incorporate probability distribution approximations.

Markov and probability distribution approximation methods

An attractive approach from the theoretical point of view is through sequential compound decision theory,[2,93,136,157] in which a decision is usually made on one character at a time but using information from the entire past. In other words, to make a decision on the n-th character in the text a decision is made on the class, or value of θ_n, which maximizes the *a posteriori* probability $P(\lambda_n = \theta_n \mid X_1, X_2, \ldots, X_n)$. These methods also assume that the language is an m-th order Markov source and hence no dictionary is needed. For example, if $m = 1$ only the so-called *bigram transition probabilities* $P(\lambda_n = \theta_n \mid \lambda_{n-1} = \theta_{n-1})$ are needed. If $m = 2$ only the *trigram transition probabilities* $P(\lambda_n = \theta_n \mid \lambda_{n-1} = \theta_{n-1}, \lambda_{n-2} = \theta_{n-2})$ are needed, and so on. Under a first-order Markov assumption ($m = 1$) and assuming conditional independence among the feature vectors, Raviv[136] and Abend[2] showed that the *a posteriori* probability at the n-th stage (n-th character in the sequence or text) can be expressed as a function of known data and the *a posteriori* probability at the $(n-1)$st stage, i.e. it can be recursively computed, and hence storage does not grow with n. Let θ_i take on any of the 26 values corresponding to the letters of the alphabet denoted by $\theta_i \in \{1,2,\ldots,26\}$, $i = 1,2,\ldots,N$, where N is the number of characters in the text. Raviv[136] showed that at stage n the posterior probability for class k $P(\lambda_n = k \mid X_1, X_2, \ldots, X_n)$ is related to $P(\lambda_{n-1} = i \mid X_1, X_2, \ldots, X_{n-1})$ $i = 1,2,\ldots,26$ (the posterior probabilities at stage $n-1$) as follows:

$$P(\lambda_n = k \mid X_1, X_2, \ldots, X_n) = \frac{P(X_n \mid \lambda_n = k) \sum_{i=1}^{26} P(\lambda_{n-1} = i \mid X_1, X_2, \ldots, X_{n-1}) P(\lambda_n = k \mid \lambda_{n-1} = i)}{\sum_{j=1}^{26} P(X_n \mid \lambda_n = j) \sum_{i=1}^{26} P(\lambda_{n-1} = i \mid X_1, X_2, \ldots, X_{n-1}) P(\lambda_n = j \mid \lambda_{n-1} = i)} \quad (5)$$

where the terms $P(\lambda_n = k \mid \lambda_{n-1} = i), k, i = 1,2,\ldots,26$ are the bigram transition probabilities. The summation signs take care of the information contained in the previous (stage 1 to stage $n-1$) feature vectors that is supposedly relevant to the decision at stage n. While this is an elegant solution and storage does not grow with n, one wonders whether the computation involved in the summation signs is worth the effort in terms of error correction improvement. In particular, if we assume that the decision made at stage $n-1$ is correct, and use that decision (call it $D_{n-1} \in \{1,2,\ldots,26\}$) instead of the previous feature vectors, this is equivalent to assuming that

$$P(\lambda_{n-1} = i \mid X_1, X_2, \ldots, X_{n-1}) = 1 \text{ for } i = D_{n-1}$$

and

$$P(\lambda_{n-1} = i \mid X_1, X_2, \ldots, X_{n-1}) = 0 \text{ for } i \neq D_{n-1}.$$

Instead of maximizing equation (5), we now maximize $P(\lambda_n = k \mid X_n, D_{n-1})$ which is equivalent to maximizing

$$P(X_n \mid \lambda_n = k) P(\lambda_n = k \mid \lambda_{n-1} = D_{n-1}), \quad (6)$$

which is much simpler than (5). This version of (5) was in fact mentioned by Raviv, but no experimental results were reported by him. Hanson *et al.*[81] compared (6) with classification without using context, i.e.

with a classifier which chooses k such that $P(X_n \mid \lambda_n = k)P(\lambda_n = k)$ is maximum, and reported that use of (6) resulted in an actual increased error rate over not using context.

Extensive comparison of (6) both with Raviv's algorithm and with classification without context, in which experiments were done with large passages of text under widely varying conditions, showed results contrary to Hanson's. In fact, not only did (6) reduce the error rate over classification without context, but, surprisingly, yielded virtually the same results as Raviv's algorithm. These results make us conclude that when using statistical contextual information contained in the past patterns to classify the present pattern, given the present feature vector, no significant additional information is contained in the past feature vectors over the decision of the immediately preceding pattern. For further comments the reader is referred to Shinghal *et al.* This is not to say, however, that measurement information from past patterns may not be preferred to additional measurement information about the present pattern to be classified, when we are dealing with a sequential classifier. Hussain has studied extensively both sequential classification methods (measurements observed sequentially), sequential compound decision theoretic approaches to using context, and their combination. Hussain generalized Wald's SPRT and proposed the compound sequential probability ratio test (CSPRT) to handle contextual information.[90] Furthermore, he showed, both theoretically and experimentally, that after k features have been observed in the sequential classification process of a pattern at stage n, the $(k+1)$st feature on the pattern at stage $n-1$ may be preferred to an additional measurement on the pattern at stage n.[92]

All the above applications of the recursive Bayes algorithm deal with text recognition where the characters are either machine-printed or handprinted. However, similar Markov models are also being applied to speech recognition.[11,96]

A second approach to approximating the distribution of a sequence of characters and combatting the search problem is through sequential decoding algorithms using stacks. Not enough is known experimentally of these algorithms in order to meaningfully compare them to others outlined here.

A third approach consists of proceeding in blocks of N characters at a time (not necessarily words) and realizing efficiency in storage and computation by (1) approximating the N-gram probability by a suitable approximation of lower-order marginals using techniques such as those outlined in the second section on image classification, and (2) defining a *depth of search* parameter d to consider only the d most likely alternatives (in some sense) to each character. This approach is more general than the Markov approach since the Markov assumption reduces the N-gram to one type of *product approximation* which is itself a special kind of *extension approximation.* An example will clarify the distinction. To test a three-letter sequence such as THE would involve the probability $P(\lambda_1 = T, \lambda_2 = H, \lambda_3 = E)$. In order to use bigrams rather than trigrams, and save considerably on storage, the Markov approach dictates that we should use

$$P(\lambda_3 = E \mid \lambda_2 = H)P(\lambda_2 = H \mid \lambda_1 = T)P(\lambda_1 = T).$$

In general we can use other approximations to $P(\lambda_1 = T, \lambda_2 = H, \lambda_3 = E)$. It was shown[178,179] that using the approximation $P(\lambda_1 = T, \lambda_2 = H)P(\lambda_2 = H, \lambda_3 = E)$, which does not even qualify as a product approximation, gives better results than assuming Markov dependence and accounts for 80% of the improvement in going from bigrams to trigrams.

Another elegant Markov technique in which a decision is made on one word at a time (although this is not necessary) is through the use of the Viterbi algorithm. This approach to using context was initially proposed and investigated by Forney,[63,64] and Neuhoff.[125] The Viterbi algorithm provides an efficient way of finding the most likely sequence of characters in a maximum *a posteriori* probability sense. The Viterbi algorithm is illustrated by example in Fig. 14 where we assume, for simplicity, a four-letter alphabet, i.e. $\theta \in \{T, A, H, O\}$. Let us assume that the machine is fed the word –HAT–. where "–" denotes blank. The feature extractor "looks" at H, A, T and obtains, respectively, the feature vectors X_1, X_2, and X_3, which it relays on to the classifier. The information available and relevant to making a decision on the word can be expressed in terms of a directed graph or trellis as in Fig. 14. The directed graph consists of nodes and directed edges. All nodes (except the "start" node and "end" node identified by the blank character "–") and edges have probabilities associated with them. The *edge probabilities* represent *static information*; that is, they remain fixed no matter what sequence of letters is presented to the machine. In fact, these probabilities are the Markov transition probabilities between letters. The *node probabilities*, on the other hand, represent *dynamic information*; that is, they are a function of the actual characters presented to the machine. In fact, these probabilities are the likelihoods of the feature vectors obtained from the characters. We see from Fig. 14 that any path from the start node to the end node represents a sequence of letters but not necessarily a valid word. Consider the dark path in Fig. 14. If we add the logarithms of all the edge and node probabilities encountered on that path, we obtain

$$\begin{aligned} g_{HAT}(X_1, X_2, X_3) = {} & \log P(X_1 \mid H) + \log P(X_2 \mid A) \\ & + \log P(X_3 \mid T) \\ & + \log [P(H \mid -)P(A \mid H)P(T \mid A)P(- \mid T)]. \end{aligned} \tag{7}$$

Returning for a moment to equation (3), we see that if a first-order Markov assumption is invoked in (3), then (3) reduces to (7). Hence, the Viterbi algorithm does not implement a fundamentally different decision than does (3); rather, it is an efficient method of searching for the path in the graph having the maximum product

of path probabilities, and thus, of making the same decision as (3) when the Markov assumption is invoked. In the 26-letter alphabet of the English language this efficiency is obtained as follows. At each stage in the sequence (word), 26 most likely sequences are computed: the most likely sequence ending in *A*, the most likely sequence ending in *B*, etc. At the final stage the most likely sequence is chosen.

The Viterbi algorithm can also be used with a confusion matrix rather than the likelihood probabilities and hence it also falls under the class of spelling-correction algorithms. A comparison between the two modes of operating the Viterbi algorithm will appear in Toussaint and Chung.[175]

We have proposed a modified Viterbi algorithm in which a depth of search parameter limits the number of alternatives considered for each character. Extensive experimentation with the modified Viterbi algorithm can be found in Chung,[39] Shinghal,[155] and Shinghal, Rosenberg and Toussaint.[184] The modified Viterbi algorithm gives the same performance as the Viterbi algorithm and requires far less computation. From all our experience with the various Markov approaches we conclude that the uniformly best overall method is the modifiedViterbi algorithm. In spite of this, the modified Viterbi method is not outstanding in error correction capability. In fact all Markov methods can be characterized as requiring little storage and computation but exhibiting mediocre error correction capability. This is in contrast to dictionary look-up methods which correct errors very well but at the price of excessive computation and storage. An obvious step at this point is to investigate combined methods to try to find those that have the good qualities of both approaches. These techniques are referred to as hybrid methods and are treated in the following section.

Hybrid methods

Hybrid methods use a dictionary as well as statistics such as bigrams. This means that they require storing the dictionary and the question becomes: can computation, such as that in the Bledsoe–Browning approach be significantly reduced, so that these methods become attractive? The answer is yes, but not much has been done in this area as it has become of interest only recently. One hybrid approach was considered by Riseman and Ehrich, 1971. They proposed using bigrams to eliminate from consideration and dictionary searching a large number of the sequences formed from the alternatives. They propose using word-position dependent bigrams quantized into two values 0 or 1. For example, the word $T_1H_2A_3T_4$, where the subscript denotes position in the word, would involve such bigrams as $P(T_1H_2)$, $P(T_1A_3)$, etc. If one such bigram for a string or sequence is 0, the string is not searched in the dictionary. This basic idea has recently been generalized in several directions.[58–60, 62, 80, 81, 137, 138, 139]

Another hybrid approach recently investigated by Shinghal[157] is a true combination of the modified Viterbi algorithm and the dictionary look-up approach of Bledsoe and Browning. The method is referred to as the predictor–corrector algorithm and is illustrated in Fig. 15. The dictionary is partitioned into blocks of words of equal length and the words are sorted according to a VALUE as shown in the figure. In step 1, "$d = 2$" indicates the depth-of-search parameter, i.e. only the two most likely alternatives for each letter in the word are considered. In steps 4 and 5 the "score" is a discriminant function as in (7). This algorithm reduces the cost of the dictionary look-up algorithm by 50% without a noticeable increase in

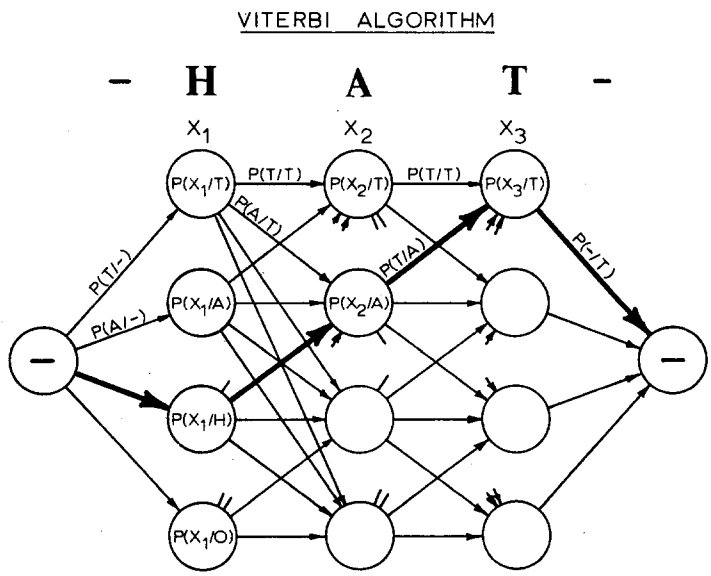


Fig. 14.

error rate. As described in Fig. 15 the algorithm does not require calculating and sorting by VALUE to organize the dictionary. Sorting of words could be done alphabetically and step 2 deleted. Step 3 would then use binary search to determine whether Z_n is in the DICTIONARY. The actual algorithm is more complicated and uses VALUE(Z_n) to compute scores in step 4 for only a subset of the words of length n. Details and results on this algorithm will appear in Toussaint and Shinghal.[186] As a final note, it is worth pointing out that this approach is a combined *bottom-up* and *top-down* approach to context as was discussed in the introduction. Taking dependencies between letters in the Viterbi algorithm represents the *bottom-up* portion. Using the dictionary to constrain the selection of letter sequences to valid words represents the *top-down* aspect. Hence, this algorithm represents a formal model of the bottom-up-top-down theoretical construct being presently considered in information processing models of human behaviour.

MISCELLANEOUS ASPECTS OF CONTEXT

There are several aspects of context which do not fit nicely into the categories discussed above. One is the problem of finding theoretical bounds on the probability of error when context is used. Some work in this area can be found in Chen,[29] Chu,[37,38] Fisher,[62] Neuhoff,[125] and Toussaint.[179] Unsupervised learning approaches to context can be found in Casey and Nagy,[26] and Hilborn and Lainiotis.[88,89] The application of contextual information to machine recognition of hand sent Morse code can be found in Gold[74] and McElwain and Evens.[114]

PREDICTOR - CORRECTOR ALGORITHM

DICTIONARY ORGANIZATION

n=2 n=3 n=4

WORDS	VALUE
THE	-18.39
ATE	-19.20
ARE	-19.77
TIN	-19.82
TEN	-20.05
ONE	-20.31
•	•
•	•
•	•

$$\text{VALUE(HAT)} = \log\left[P(H/-)\,P(A/H)\,P(T/A)\,P(-/T)\right]$$

STEP 1: Use modified Viterbi algorithm (d=2) to determine output sequence Z_n.

STEP 2: Compute VALUE(Z_n).

STEP 3: Use binary search to determine whether VALUE(Z_n) is in DICTIONARY; if YES exit with Z_n; if NO continue.

STEP 4: Compute scores for all words of length n in DICTIONARY.

STEP 5: Select word with highest score.

Fig. 15.

One area of research which provides a theoretical foundation for understanding context, and also provides experimental guidelines, is the information theoretic study of the redundancy of language. A classic paper in this area is the one by Shannon.[149] Since then various approaches have been taken to obtain estimates of the redundancy of language.[13, 22, 23, 25, 28, 43, 77, 111, 126, 156]

A question of obvious interest is: given the nth letter in a piece of text, is it significantly constrained by letters in the distant past. Alternately, what is the basic unit of contextual information. Researchers in different fields give different answers. Most of the work cited in this paper obviously is considering the "word" as the basic unit. The answers also depend upon whether one is talking about statistical or semantic contextual information. Burton and Licklider conclude that while the estimate of relative redundancy increases as knowledge of the foregoing text is extended from zero to approximately 32 letters, increasing the known number of letters beyond 32 does not result in any notable rise. Classification experiments however indicate that considering dependencies beyond the 2 previous letters (trigrams) does not produce a notable decrease in error rate.[56] Some researchers believe that the basic unit of meaning is, not the sentence or, even, paragraph but, the *situational frame*, which is a cultural unit.[79] The importance and relation of culture to context is very well discussed by Hall. Other aspects of culture and context with respect to pattern recognition can be found in Deregowski[49] and Segall *et al.*[147,148] Finally, for a discussion of the limits imposed on artificial intelligence by context, culture, and pattern recognition, the reader is referred to Toussaint.[183]

SUMMARY

The importance of context at the perceptual, cognitive, and "objective" mathematical levels is discussed and illustrated with examples. Several concepts such as disambiguation, error-correction, and filling in the gaps are also discussed from the pattern recognition point of view. A tutorial survey of techniques for using contextual information in pattern recognition follows. The emphasis is on image classification and text recognition, where the text is in the form of machine and handprinted characters, cursive script, and speech. One of the main observations from the literature on using context in image classification and text recognition is that in order to use statistical contextual information in an efficient manner, it is necessary to use suitable approximations to otherwise unmanageable probability distributions. The techniques for using context in text recognition are classified into three categories: dictionary look-up methods, Markov and probability distribution approximation methods, and hybrid methods. All the approaches are compared under the unifying theme of compound decision

theory. Finally, some miscellaneous aspects of context are referenced and briefly discussed.

REFERENCES

1. K. Abend, T. J. Harley and L. N. Kanal, Classification of binary random patterns, *IEEE Trans. Inf. Theory* **11,** 538–544 (1965).
2. K. Abend, Compound decision procedures for unknown distributions and for dependent states of nature, in *Pattern Recognition* (ed. L. N. Kanal). Thompson, Washington, D.C., pp. 204–249 (1968).
3. J. L. Adams, *Conceptual Blockbusting.* San Francisco Book Co. (1976).
4. C. N. Alberga, String similarity and misspellings, *CACM* **10,** 302–313 (1967).
5. R. Alter, Utilization of contextual constraints in automatic speech recognition, *IEEE Trans. Audio Electroacoust.* **16,** 6–11 (1968).
6. R. Arnheim, *Visual Thinking.* University of California Press (1969).
7. R. Arnheim, *Art and Visual Perception.* University of California Press (1974).
8. A. D. Baddeley, R. Conrad and W. E. Thomson, Letter structure of the English language, *Nature* **186,** 414–416 (1960).
9. R. R. Bahadur, A representation of the joint distribution of responses to *n* dichotomous items, in *Studies in Item Analysis and Prediction* (ed. H. Solomon). Stanford University Press, Stanford, CA (1961).
10. L. R. Bahl and F. Jelinek, Decoding for channels with insertions, deletions, and substitutions with applications to speech recognition, *IEEE Trans. Inf. Theory* **21,** 404–411 (1975).
11. J. K. Baker, Stochastic modeling for automatic speech understanding, in *Speech Recognition* (ed. D. Raj Reddy). Academic Press, New York, pp. 73–82 (1975).
12. Y. Bar-Hillel, A demonstration of the nonfeasibility of fully automatic high quality translation, in *Advances in Computers,* Vol. I (ed. F. L. Alt). Academic Press, New York, pp. 158–163 (1960).
13. C. D. Basharin, On a statistical estimate for the entropy of a sequence of independent random variables, *Theory Probab. Applic.* **4,** 333–336 (1959).
14. P. W. Becker, What multivariate probability densities are possible with a specified set of marginals, *Proc. 2nd Int. Jnt Conf. on Pattern Recognition,* 13–15 August, Copenhagen, pp. 368–371 (1974).
15. P. Billingsley, Statistical methods in Markov chains, *Ann. Math. Statist.* **32,** 12–40 (1961).
16. C. R. Blair, A program for correcting spelling errors, *Inf. Control* **3,** 60–67 (1960).
17. W. W. Bledsoe and I. Browning, Pattern recognition and reading by machine, *Proc. Eastern Jnt Computer Conf.* **16,** pp. 225–232 (1959).
18. B. Blesser, A theoretical approach for character recognition based on phenomenological attributes, *Proc. 1st Int. Jnt Conf. on Pattern Recognition,* Washington, D.C., pp. 33–40, October (1973).
19. R. Bornat and J. M. Brady, Using knowledge in the computer interpretation of handwritten FORTRAN coding sheets, *Int. J. Man–Machine Stud.* **8,** 13–27 (1976).
20. C. P. Bourne and D. F. Ford, A study of the statistics of letters in English words, *Inf. Control* **4,** 48–67 (1961).
21. D. T. Brown, A note on approximations to discrete probability distributions, *Inf. Control* **2,** 386–392 (1959).
22. J. S. Bruner and D. O'Dowd, A note on the informativeness of parts of words, *Language and Speech* **1,** 98 – 101 (1958).
23. N. C. Burton and J. C. R. Licklider, Long-range constraints in the statistical structure of printed English, *Am. J. Psychol.* **68,** 650–653 (1955).
24. G. Carlson, Techniques for replacing characters that are garbled on input, *Proc. Spring Jnt Computer Conf.* **30,** 189–192 (1966).
25. D. H. Carson, Letter constraints within words in printed English, *Kybernetik* **1,** 46–54 (1961).
26. R. G. Casey and G. Nagy, An autonomous reading machine, *IEEE Trans. Comput.* **17,** 492–503 (1968).
27. S. K. Chang and G. Nagy, Phonetic string retrieval using imperfectly labeled prototypes, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition,* Coronado, pp. 632–637 (1976).
28. C. Chatfield, Statistical inference regarding Markov chain models, *Appl. Statist.* **22,** 7–20 (1973).
29. C. H. Chen, *Statistical Pattern Recognition.* Hayden Book Co. (1973).
30. C. K. Chow, An optimum character recognition system using decision functions, *IRE Trans. electr. Comput.* **6,** 247–254 (1957).
31. C. K. Chow, A recognition method using neighbour dependence, *IEEE Trans. Comput.* **11,** 683–690 (1962).
32. C. K. Chow, A class of nonlinear recognition procedures, *IEEE Trans. Syst., Sci., Cybern.* **2,** 101–109 (1966).
33. C. K. Chow and C. N. Liu, An approach to structure adaptation in pattern recognition, *IEEE Trans. Syst., Sci., Cybern.* **2,** 73–80 (1966).
34. C. K. Chow, The maximum likelihood estimate of a dependence tree, in *Pattern Recognition* (ed. L. N. Kanal). Thompson Book Co., pp. 323–328 (1968).
35. C. K. Chow and C. N. Liu, Approximating discrete probability distributions with dependence trees, *IEEE Trans Inf. Theory* **14,** 462–467 (1968).
36. C. S. Christensen, An investigation of the use of context in character recognition using graph searching. Cornell University, Report submitted to Air Force Office of Scientific Research (1968).
37. J. T. Chu, Error bounds for a contextual recognition procedure, *IEEE Trans. Comput.* **20,** 1203–1207 (1971).
38. J. T. Chu, Author's reply [to Toussaint, 1972], *IEEE Trans. Comput.* **21,** 1027–1028 (1972).
39. S. S. Chung, Using contextual constraints from the English language to improve the performance of character recognition machines, M.Sc. thesis, School of Computer Science, McGill University (1975).
40. G. Cohen and M. Martin, Hemisphere differences in an auditory Stroop test, *Percept. Psychophys.* **17,** 79–83 (1975).
41. R. W. Cornew, A statistical method of spelling correction, *Inf. Control* **12,** 79–93 (1969).
42. T. M. Cover, Thes best two independent measurements are not the two best, *IEEE Trans. Syst., Man, Cybern.* **4,** 116–117 (1974).
43. T. M. Cover and R. C. King, A convergent gambling estimate of the entropy of English language, *IEEE Trans. Inf. Theory,* to be published.
44. C. Cox, B. Blesser and M. Eden, The application of type font analysis to automatic character recognition, *Proc. 2nd Int. Jnt Conf. on Pattern Recognition,* Copenhagen, Denmark, 13–15 August, pp. 226–232 (1974).
45. F. J. Damerau, A technique for computer detection and correction of spelling errors, *CACM* **7,** 171–176 (1964).
46. L. Davidson, Retrieval of misspelled names in an airlines passenger record system, *CACM* **5,** 169–171 (1963).
47. P. B. Denes, The design and operation of a mechanical speech recognizer at University College London, *J. Br. Inst. Radio Eng.* **19,** 219–229 (1959).
48. P. B. Denes, Automatic speech recognition: old and new ideas, in *Speech Recognition* (ed. D. Raj Reddy). Academic Press, New York, pp. 73–82 (1975).
49. J. B. Deregowski, Illusion and culture, in *Illusion in Nature and Art* (eds. R. L. Gregory and E. H. Gom-

brich). Duckworth & Co., London, pp. 161–191 (1973).
50. R. W. Donaldson and G. T. Toussaint, Use of contextual constraints in recognition of contour traced handprinted characters, *IEEE Trans. Comput.* **19,** 1095–1099 (1970).
51. W. Doster, Contextual postprocessing system for cooperation with a multiple choice character recognition system, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition*, Coronado, pp. 653–657 (1976).
52. H. Dreyfus, *What Computers Can't Do.* Harper & Row, New York (1972).
53. R. O. Duda and P. E. Hart, Experiments in the recognition of handprinted text – II. Context analysis, *Proc. 1968 Fall Jnt Comput. Conf.* **34,** pp. 1139–1149 (1968).
54. R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis.* Wiley, New York (1973).
55. F. N. Dyer, The Stroop phenomenon and its use in the study of perceptual, cognitive, and response processes, *Memory and Cognition* **1,** 106–120 (1973).
56. C. A. Dykema, A computer simulation study of the application of contextual constraints to character recognition, M.A.Sc. thesis, University of British Columbia, September (1970).
57. A. W. Edwards and R. L. Chambers, Can *a priori* probabilities help in character recognition? *JACM* **11,** 465–470 (1964).
58. R. W. Ehrich, A contextual post processor for cursive script recognition – summary, *Proc. 1st Int. Jnt Conf. on Pattern Recognition*, Washington, D.C., pp. 169–171 (1973).
59. R. W. Ehrich and K. J. Koehler, Experiments in the contextual recognition of cursive script, *IEEE Trans. Comput.* **24,** 182–194 (1975).
60. R. W. Ehrich and E. M. Riseman, Contextual error detection, *COINS Tech. Rpt*, 700-4, University of Massachusetts, Amherst (1971).
61. J. Fang, Towards a certain "Contextualism": (foresight vs hindsight) vs insight, *Phil. Math.* **9,** 158–167 (1972).
62. E. G. Fisher, The use of context in character recognition, *COINS Tech. Rpt 76-12*, Computer Science, University of Massachusetts at Amherst (1976).
63. G. D. Forney, The Viterbi algorithm, *Proc. IEEE* **61,** 268–278 (1973).
64. G. D. Forney, Jr., MAP sequence estimation of Markov sequences via the Viterbi algorithm, with application to text recognition, unpublished paper, Stanford Electronics Lab., August (1971).
65. J. J. Forsyth, Error probability in dependent pattern classification, *IEEE Tans. Inf. Theory*, 678–680 (1972).
66. J. J. Freeman, Note on approximating discrete probability distributions, *IEEE Trans. Inf. Theory* **17,** 491–492 (1971).
67. M. B. Freeman, A theorem on extensions of discrete probability distributions; Report No. 60-RL-2529E. General Electric Lab. (1960).
68. L. S. Frishkopf, and L. D. Harmon, Machine reading of cursive script, in *Information Theory* (ed. C. Cherry) pp. 300–316. Butterworth, London (1961).
69. K. S. Fu, *Syntactic Methods in Pattern Recognition.* Academic Press, New York (1974).
70. L.-W. Fung and K.-S. Fu, Stochastic syntactic decoding for pattern classification, *IEEE Trans. Comput.* **24,** 662–667 (1975).
71. J. J. Giangardella, J. F. Hudson and R. S. Roper, Spelling correction using a digital computer, *IEEE Trans. Engng Writing Speech* **10,** 57–62 (1967).
72. E. J. Gibson, Perceptual learning and the theory of word perception, *Cogn. Psychol.* **2,** 351–368 (1971).
73. H. T. Glantz, On the recognition of information with a digital computer, *JACM* **4,** 178–188 (1957).
74. B. Gold, Machine recognition of hand sent morse code, *IRE Trans. Inf. Theory* **5,** 17–24 (1959).
75. R. H. Granger, FOUL-UP: A program that figures out meanings of words from context, *Proc. 5th Int. Jnt Conf. on Artificial Intelligence*, 22–25 August, M.I.T., Cambridge, MA (1977).
76. P. D. Green and W. A. Ainsworth, Towards the automatic recognition of spoken basic English, *Proc. Machine Perception of Patterns and Pictures*, pp. 161–168 (1972).
77. M. Grignetti, A note on the entropy of words in printed English, *Inf. Control* **7,** 304–306 (1964).
78. A. Guzman, Analysis of curved line drawings using context and global information, in *Machine Intelligence* **6** (eds. B. Meltzer and D. Michie). Edinburgh (1971).
79. E. T. Hall, *Beyond Culture.* Anchor Press, New York (1976).
80. A. R. Hanson and E. H. Riseman, System design of an integrated pattern recognition system or how to get the best mileage out of your used pattern classifier, *COINS Tech. Rt* 73C-5, Univ. of Massachusetts, Amherst, June (1973).
81. A. R. Hanson, E. M. Riseman and E. Fisher, Context in word recognition, *Pattern Recognition* **8,** 35–45 (1976).
82. L. D. Harmon, Automatic reading of cursive script, in *Optical Character Recognition* (eds. Fischer, Pollock, Radack and Stevens) pp. 151–152. Spartan Books, Washington, D.C. (1962).
83. L. D. Harmon, Automatic recognition of print and script, *Proc. IEEE* **60,** 1165–1176 (1972).
84. L. D. Harmon and E. J. Sitar, Method and apparatus for correcting errors in mutilated text, *U.S. Patent 3 188 609*, issued 8 June (1965).
85. J. Hartmanis, The application of some basic inequalities for entropy, *Inf. Control* **2,** 199–213 (1959).
86. R. C. Heinselman, Computerized detection and correction of spelling errors in FORTRAN programs, M.Sc. thesis, Department of Information Science, University of Minnesota, Minneapolis (1972).
87. R. B. Hennis, The IBM 1975 optical page reader, part I–system design, *IBM J. Res. Dev.* **12,** 346–353 (1968).
88. C. G. Hilborn, Jr. and D. G. Lainiotis, Unsupervised learning minimum risk pattern classification for dependent hypotheses and dependent measurements, *IEEE Trans. Syst., Sci., Cybern.* **5,** 109–115 (1969).
89. C. G. Hillborn, Jr. and D. G. Lainiotis, Optimal unsupervised learning multicategory dependent hypotheses pattern recognition, *IEEE Trans. Inf. Theory* **14,** 468–470 (1968).
90. A. B. S. Hussain, Compound sequential probability ratio test for the classification of statistically dependent patterns, *IEEE Trans. Comput.* **23,** 398–410 (1974).
91. A. B. S. Hussain, Sequential methods in statistical pattern recognition, Ph.D. thesis, University of British Columbia, Vancouver (1972).
92. A. B. S. Hussain and R. W. Donaldson, Some Sequential sampling techniques for multicategory dependent hypotheses pattern recognition, *Proc. 5th Hawaii Int. Conf. on System Sciences*, pp. 40–43 (1972).
93. A. B. S. Hussain, G. T. Toussaint and R. W. Donaldson, Results obtained using a simple character recognition procedure on Munson's handprinted data, *IEEE Trans. Comput.* **21,** 201–205 (1972).
94. J. Huttenlocher and S. L. Weiner, Comprehension of instructions in varying contexts, *Cogn. Psychol.* **2,** 369–385 (1971).
95. C. T. Ireland and S. Kullback, Contingency tables with given marginals, *Biometrika* **55,** 179 (1968).
96. F. Itakura, Minimum prediction residual principle applied to speech recognition, *Proc. IEEE Symposium on Speech Recognition*, Pittsburgh, PA, 101–105 (1974).
97. F. Jelinek, The design of a linguistic statistical decoder for the recognition of continuous speech, *IEEE Trans. Inf. Theory* **21,** 250–256 (1975).
98. J. J. Jenkins, Remember that old theory of memory?

Well, forget it! *Am. Psychol.* **29,** 785–795, November (1974).
99. B. Jones and O. Brigham, A note on the determination of a discrete probability distribution from known marginals, *Inf. Control* **15,** 527–528 (1969).
100. B. Jones, Relating and approximating discrete probability distributions with component distributions, *Inf. Sci.* **6,** 143–150 (1973).
101. L. Kanal, Patterns in pattern recognition, 1968–1974, *IEEE Trans. Inf. Theory* **20,** 697–722 (1974).
102. R. L. Kashyap and M. C. Mittal, Word recognition in a multi-talker environment using syntactic methods, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition*, Coronado, pp. 626–631 (1976).
103. R. L. Kashyap and M. C. Mittal, A new method for error correction in strings with applications to spoken word recognition, *Proc. IEEE Comput. Soc. Conf. on Pattern Recognition and Image Processing*, Troy, N.Y., pp. 76–82 6–8 June (1977).
104. P. A. Kolers and D. N. Perkins, Spatial and ordinal components of form perception and literacy, *Cogn. Psychol.* **7,** 228–268 (1975).
105. H. Ku and S. Kullback, Approximating discrete probability distributions, *IEEE Trans. Inf. Theory* **15,** 444–447 (1969).
106. S. Kullback, Probability densities with given marginals, *Ann. Math. Statist.* **39,** 1236–1243 (1968).
107. M. Kuvera and W. H. Francis, *Computational Analysis of Present-Day American English.* Brown University Press, Providence, R.I. (1967).
108. P. M. Lewis, II, A note on the realization of decision networks using summation elements, *Inf. Control* **4,** 282–290 (1961).
109. M. Manis, Context effects in communication: determinants of verbal output and referential decoding, in *Adaption-Level Theory, a Symposium* (ed. M. H. Appley) Academic Press, New York (1971).
110. T. B. Martin, Applications of limited vocabulary recognition systems, in *Speech Recognition* (ed. D. Raj Reddy). Academic Press, New York, pp. 73–82 (1975).
111. V. Maxiner, Some remarks on entropy prediction of natural language, *Inf. Storage Retrieval* **7,** 293–295 (1971).
112. M. S. Mayzner and M. E. Tresselt, Tables of single-letter and diagram frequency counts for various word-length and letter-position combinations, *Psychon. Monogr. Suppl.* **1,** 13–32 (1965).
113. M. S. Mayzner, M. E. Tresselt and B. R. Wolin, Tables of trigram frequency counts for various word-length and letter-position combinations, *Psychon. Monogr. Suppl.* **1,** 33–78 (1965).
114. C. K. McElwain and M. B. Evens, The degarbler – a program for correcting machine-read morse code, *Inf. Control* **5,** 368–384 (1962).
115. P. Melmerstein and M. Eden, A system for automatic recognition of handwritten words, *Proc. Fall Jnt Computer Conf.* **26,** Part 1, pp. 333–342 (1964).
116. G. A. Miller, Decision units in the perception of speech, *IRE Trans. Inf. Theory,* **8,** 81–83 (1962).
117. G. A. Miller, The intelligibility of speech as a function of the context of the text materials, *J. exp. Psychol.* **41,** 329–335 (1951).
118. G. A. Miller and E. B. Coleman, A set of thirty-six prose passages calibrated for complexity, *J. Verbal Learning Verbal Behav.* **6,** 851–854 (1967).
119. G. A. Miller and E. A. Friedman, The reconstruction of mutilated English texts, *Inf. Control* **1,** 38–55 (1957).
120. G. A. Miller, E. B. Newman and E. A. Friedman, Length-frequency statistics for written English, *Inf. Control* **1,** 370–389 (1958).
121. H. L. Morgan, Spelling correction in systems programs, *CACM* **13,** 90–94 (1970).
122. J. H. Munson, The recognition of hand-printed text, in *Pattern Recognition* (ed. L. Kanal). Thompson Book Co., Washington, D.C., pp. 115–140 (1968).
123. B. Nash-Webber, The role of semantics in automatic speech understanding, in *Representation and Understanding* (eds D. G. Bobrow and A. Collins). Academic Press, New York, pp. 351–382 (1975).
124. U. Neisser and P. Weene, A note on human recognition of hand-printed characters, *Inf. Control* **3,** 191–196 (1960).
125. D. L. Neuhoff, The Viterbi algorithm as an aid in text recognition, *IEEE Trans. Inf. Theory* **21,** 222–226 (1975).
126. E. B. Newman and N. C. Waugh, The redundancy of texts in three languages, *Inf. Control* **3,** 141–153 (1960).
127. D. A. Norman and D. G. Bobrow, On the role of active memory processes in perception and cognition, in *The Structure of Human Memory* (ed. C. N. Cofer). Freeman, New York (1976).
128. T. Okuda, Method for the correction of garbled words, *Computer Group Repository*, R74-64, 29pp.
129. T. Okuda, An error correcting method based on the extended Hamming metric, *Proc. Jnt Meeting of Nagoya Branches of Four Electric Assoc.* (Japan), 18a-B-6, October (1972).
130. T. Okuda, E. Tanaka and K. Tamotsu, A method for the correction of garbled words based on Levenshtein metric, *IEEE Trans. Comput.* **25,** 172–177, February (1976).
131. C. E. Oleson, EXAMINER: A system using contextual knowledge for analysis of diagnostic behaviour, *Proc. 5th Int. Jnt Conf. on Artificial Intelligence*, M.I.T., Cambridge, MA, 22–25 August (1977).
132. S. E. Palmer, Visual perception and world knowledge, in *Explorations in Cognition* (eds D. A. Norman, D. E. Rumelhart, & the LNR Research Group). Freeman, New York (1975).
133. S. C. Pepper, *World Hypotheses.* University of California Press (1970).
134. J. R. Pierce, *Symbols, Signals and Noise.* Harper & Row, New York (1961).
135. F. Pratt, *Secret and Urgent.* Blue Ribbon Books, Garden City, N.J. (1942).
136. J. Raviv, Decision making in Markov chains applied to the problem of pattern recognition, *IEEE Trans. Inf. Theory* **13,** 536–551, October (1967).
137. E. M. Riseman and R. W. Ehrich, Contextual word recognition using binary digrams, *IEEE Trans. Comput.* **20,** 397–403, April (1971).
138. E. M. Riseman and A. R. Hanson, A contextual postprocessing system for error correction using binary *n*-grams, *IEEE Trans. Comput.* **23,** 480–493 (1974).
139. E. M. Riseman and A. R. Hanson, A contextual postprocessing system for error detection and correction in character recognition, *COINS Techn. Rpt* **1,** Univ. Massachusetts, Amherst (1972).
140. W. S. Rosenbaum and J. J. Hillard, Multifont OCR post-processing system, *IBM J. Res. Dev.* 398–421 (1975).
141. A. Rosenfeld, Iterative methods in image analysis, *Proc. IEEE Computer Soc. Conf. on Pattern Recognition and Image Processing*, Troy, New York, pp. 14–18, 6–8 June (1977).
142. D. E. Rumelhart and P. Siple, The process of recognizing tachistoscopically presented words, *Psychol. Rev.* **81,** 99–118 (1974).
143. J. Sachs, Recognition memory for syntactic and semantic aspects of connected discourse, *Percept. Psychophys.* **2,** 437–442 (1967).
144. T. Sakai and S. Nakagawa, Speech understanding system LITHAN – and some applications, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition*, Coronado, pp. 621–625, November (1976).
145. K. M. Sayre, Machine recognition of handwritten

words: a project report, *Pattern Recognition* **5,** 213–223 (1973).
146. J. Schurmann, Multifont word recognition system with application to postal address reading, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition*, Coronado, pp. 658–662 (1976).
147. M. H. Segall, *Influence of Culture on Visual Perception.* Indianapolis (1966).
148. M. H. Segall, G. T. Campbell and N. J. Keyovits, Cultural differences in the perception of geometric illusions, in *Perception: Selected Reading in Science and Phenomenology* (ed. Paul Tibbetts). Quadrangle Books, Chicago, pp. 333–339 (1969).
149. C. E. Shannon, Prediction and entropy of printed English, *Bell Syst. Techn. J.* **30,** 50–64 (1951).
150. H. C. Sharp, Effect of contextual constraint upon recall of verbal passages, *Am. J. Psychol.* **71,** 568–572 (1958).
151. J. N. Sheame and P. F. Leach, Some experiments with a simple word-recognition system, *IEEE Trans. Audio Electroac.* **16,** 256 (1968).
152. R. J. Shillman, Character recognition based on phenomenological attributes: theory and methods, Ph.D. thesis, M.I.T., Cambridge, MA (1974).
153. R. J. Shillman, Experimental methodologies for character recognition based on phenomenological attributes, *Proc. 2nd Int. Jnt Conf. on Pattern Recognition*, Copenhagen, pp. 195–201, August (1974).
154. M. Shimura, Recognizing machines with parametric and non-parametric learning methods using contextual information, *Pattern Recognition* **5,** 149–168 (1973).
155. R. Shinghal, Using contextual information to improve performance of character recognition machines. Ph.D. thesis, School of Computer Science, McGill University (1977).
156. R. Shinghal and G. T. Toussaint, Cluster analysis of English text, to be published.
157. R. Shinghal, D. Rosenberg and G. T. Toussaint, A simplified heuristic version of Raviv's algorithm for using context in text recognition, *Proc. 5th Int. Jnt Conf. on Artificial Intelligence*, 22–25 August. M.I.T., Cambridge, MA (1977).
158. R. Shinghal, D. Rosenberg and G. T. Toussaint, A simplified heuristic version of a recursive Bayes algorithm for using context in text recognition, to appear in *IEEE Trans. Syst., Man, Cybern.*
159. M. L. Shneier, Recognition using semantic constraints, *Proc. 5th Int. Jnt Conf. on Artificial Intelligence*, 22–25 August. M.I.T., Cambridge, MA (1977).
160. E. J. Sitar, Machine recognition of cursive script: the use of context for error detection and correction, Tech. Memorandum, Bell Labs.
161. V. Strassen, The existence of probability measures with given marginals, *Ann. Math. Statist.* **36,** 423–439 (1965).
162. A. J. Szanser, Error correcting methods in natural language processing, *Information Processing* 68 (ed. A. J. H. Morrell). North Holland, Amsterdam **11,** 1412–1416 (1968).
163. A. J. Szanser, Automatic error correction in natural languages, *Inf. Storage Retrieval* **5,** 169–174 (1970); also in *Statistical Methods in Linguistics* **6,** 52–59 (1970).
164. A. J. Szanser, Resolution of ambiguities by contextual word repetition, *Rev. Appl. Linguist.* **7,** 49–56 (1970).
165. A. J. Szanser, Automatic error correction in natural texts – 1, *Comm. Sci.* **46** (1971).
166. A. J. Szanser, Automatic error correction in natural texts – 2, *Comm. Sci.* **53** (1971).
167. E. Tanaka and T. Kasai, Sync and substitution error-correcting code design based on the Levenshtein metric, *IEEE Trans. Inf. Theory*, to appear.
168. E. Tanaka and T. Kasai, A correcting method of garbled languages using ordered key letters, *Trans. Inst. Electr. Comm. Engng Japan* **55-D,** 363–370 (1972).
169. E. Tanaka, T. Kasai and M. Fujino, A correcting method of garbled languages using language information, *Trans. Inst. Electr. Comm. Engng* **54-C,** 294–301 (1971).
170. C. C. Tappert, Application of sequential decoding for converting phonetic to graphemic representation in automatic recognition of continuous speech, *Proc. Int. Conf. Speech Comm. and Processing*, Boston, MA, pp. 322–326, April (1972).
171. R. B. Thomas and M. Kassler, Character recognition in context, *Inf. Control* **10,** 43–64 (1967).
172. R. A. Thompson, Language correction using probabilistic grammars, *I.E.E.E. Trans. Comput.* **25,** 275–286 (1976).
173. L. E. Thorelli, Automatic correction of errors in text, *BIT* **2,** 45–62 (1962).
174. G. T. Toussaint, Recent progress in statistical methods applied to pattern recognition, *Proc. 2nd Int. Conf. on Pattern Recognition*, Copenhagen, pp. 478–489, August (1974).
175. G. T. Toussaint and S. Chung, The modified Viterbi algorithm as an aid to text recognition. Manuscript in preparation, School of Computer Science, McGill University, Montreal.
176. G. T. Toussaint, The efficient use of context in machine recognition of handprinted text, *2nd Ann. Computer Science Conf.*, Detroit, MI, abstract in Conf. Prog. p. 62. 12–14 Feb. (1974).
177. G. T. Toussaint and S. Chung, On some algorithms for using context in machine recognition of handprinted text, *2nd Ann Computer Science Conf.*, Detroit, MI, abs. in p. 62. 12–14 Feb. (1974).
178. G. T. Toussaint and R. W. Donaldson, Some simple contextual decoding algorithms applied to recognition of handprinted text, *Proc. Ann. Canadian Computer Conf.*, Session '72, Montreal, Canada, 1–3 June, pp. 422101–422115 (1972).
179. G. T. Toussaint, Comments on "error bounds for contextual recognition procedure", *IEEE Trans. Comput.* **21,** 1027 September (1972).
180. G. T. Toussaint, Feature evaluation criteria and contextual decoding algorithms in statistical pattern recognition, Ph.D. thesis, University of British Columbia (1972).
181. G. T. Toussaint, Note on optimal selection of independent binary-valued features for pattern recognition, *IEEE Trans. Inf. Theory* **17,** 618 September (1971).
182. G. T. Toussaint, Machine recognition of independent and contextually constrained contour-traced handprinted characters, M.A.Sc. thesis, University of British Columbia (1970).
183. G. T. Toussaint, Artificial intelligence and anthropology, to be published.
184. G. T. Toussaint, The use of context in pattern recognition, *Proc. IEEE Computer Soc. Conf. on Pattern Recognition and Image Processing*, Troy, New York, pp. 1–10. 6–8 June (1977).
185. G. T. Toussaint and R. W. Donaldson, Algorithms for recognizing contour-traced handprinted characters, *IEEE Trans. Comput.* **19,** 541–546 (1970).
186. G. T. Toussaint and R. Shinghal, A bottom-up and top-down approach to using context in text recognition, manuscript in preparation.
187. G. T. Toussaint, The use of context in text recognition, *Canad. Soc. for Computational Studies of Intelligence Newsletter* **1,** 21–22 August (1976).
188. J. R. Ullman, *Pattern Recognition Techniques.* Crane, Russack & Co. (1973).
189. R. Vives and J. Y. Gresser, A similarity index between strings of symbols: application to automatic word and language recognition, *Proc. 1st Int. Jnt Conf. on Pattern Recognition*, Washington, D.C., pp. 308–317 October (1973).
190. C. M. Vossler and N. M. Branston, The use of context

for correcting garbled English text, *Proc. ACM 19th National Conf.* pp. D2 4-1 to D2 4-13 (1964).
191. M. A. Wanar, A. I. Zayed, M. M. Shaker and E. H. Taha, First-second-and-third-order entropies of Arabic text, *IEEE Trans. Inf. Theory* **22,** 123 January (1976).
192. R. M. Warren and R. P. Warren, Auditory illusions and confusions, *Sci. Amer.*, December (1970).
193. D. D. Wheeler, Processes in word recognition, *Cogn. Psychol.* **1,** 59–85 (1970).
194. W. A. Woods, Syntax, semantics, and speech, in *Speech Recognition* (ed. D. Raj Reddy). Academic Press, New York, pp. 345–400 (1975).
195. S. W. Zucker, Relaxation labelling, local ambiguity, and low-level vision, in *Pattern Recognition and Artificial Intelligence.* Academic Press, New York, pp. 593–615 (1976).
196. S. W. Zucker, Relaxation labelling and the reduction of local ambiguities, *Proc. 3rd Int. Jnt Conf. on Pattern Recognition*, Coronado, November (1976).

About the Author – GODFRIED T. TOUSSAINT obtained a B.Sc. from the University of Tulsa and an M.A.Sc. and Ph.D. from the University of British Columbia in 1968, 1970, and 1972 respectively, all in Electrical Engineering. Since then he has been teaching and doing research in the areas of statistics, information theory, pattern recognition, image processing, and artificial intelligence in the School of Computer Science, McGill University, except for the summers of 1975 and 1977 which he spent in the Information Systems Laboratory, Stanford University. He is on the Advisory Board of the Journal of Structure Classification, and is Associate Editor of the Plenum Press Series on Advanced Applications in Pattern Recognition. His academic interests at the moment are directed mainly to the writing of a book, *The Art and Science of Pattern Recognition.*