State of the Art in Pattern Recognition

GEORGE NAGY, MEMBER, IEEE

Abstract—**This paper reviews statistical, adaptive, and heuristic techniques used in laboratory investigations of pattern recognition problems. The discussion includes correlation methods, discriminant analysis, maximum likelihood decisions, minimax techniques, perceptron-like algorithms, feature extraction, preprocessing, clustering, and nonsupervised learning. Two-dimensional distributions are used to illustrate the properties of the various procedures. Several experimental projects, representative of prospective applications, are also described.**

I. Perspective

PERHAPS the reason why relatively few reviews are published in the domain of pattern recognition is that prospective reviewers realize at the outset that pattern recognition is hardly a cohesive discipline in its own right. At best, it is a vast collection of highly varied problems. Any technique which contributes to the solution of any of these problems can therefore be considered as part and parcel of pattern recognition.

Vocabulary and notation, and, here and there, useful ideas, have been contributed by the following disciplines: statistical decision theory [Chow '57], switching theory [Winder '63], automata theory [Pask '60], set theory [Block '64], control theory [Widrow '64], linguistic analysis [Ledley '64], information theory [Kamentsky '64], mathematical programming [Rosen '65], and nerve net studies [Rosenblatt '62[2]]. The single reference for each item in this very incomplete list gives an example of how the "pattern recognition problem" can be formulated in terms of these disciplines. Conversely, it is relatively easy for the experienced pattern recognizer to describe almost any field of scientific or humanistic activity in terms of pattern recognition.

In an attempt to preserve some unity of viewpoint in the face of this abundance of possible angles, geometric concepts will be emphasized throughout this paper. The limits of the survey, and the no doubt tendentious organization imposed upon it, necessarily reflect only the author's bias. With this disclaimer, here is a brief rundown of what is to follow.

In most of the early work in the field it is assumed that a statistically representative data set is available for designing or training the recognition apparatus. The performance of the machine is then evaluated on a *test set* comprising samples not included in the *training set*. The design methods applicable in this context are reviewed in Section II.

In spite of the fact that completely satisfactory solutions have not yet been obtained even under the restrictive assumption of *stationarity*, there have been several attempts to readjust the parameters of a recognition system to *track* data sets with slowly changing characteristics. More generally, *unsupervised learning* refers to experiments based on training sequences of *unidentified* samples. This forms the subject matter of Section III.

The need for automatic methods to *cluster* samples according to their relative similarity first arose in numerical taxonomy, but applications for the resulting clustering techniques rapidly emerged in many areas of pattern recognition. Some simple clustering algorithms which can be used to simulate a species of self-organizing behavior are described in Section IV.

Practical considerations of computer economics often prevent the wholesale application of the methods mentioned above to real-life situations. The somewhat undignified and haphazard manipulation invoked in such cases to render the problem amenable to orderly solution is referred to variously as *preprocessing*, *filtering* or *prefiltering*, *feature* or *measurement extraction*, or *dimensionality reduction*. A distinction between these concepts and the ones mentioned earlier is that here the differences between the several classes of data under consideration are not necessarily taken into account in the selection of the parameters. Instead, the object is to provide a simplified, more economical description, which applies to all the samples. These matters are discussed at greater length in Section V.

What are all these schemes good for? Only character recognition has received widespread commercial acceptance so far. Machines are gradually upgrading their reading ability from the simple magnetic symbols used on bank checks to the somewhat less stylized numerals on credit cards and other turn-around documents, and even to typescripts in more or less arbitrary type styles. A really general-purpose page reader is still to be developed but the problem areas lie more in the realm of format than in character recognition per se. Several machines have already been marketed to read handprinted numerals in well-specified surroundings.

Speech recognition is probably the next broad area to reach commercial maturity. Hopefully the expanding vocabularies, decreasing sensitivity to variations between individual speakers, greater overall accuracy, and capability of processing longer and longer segments of continuous speech will prove irresistibly attractive to system designers preoccupied with man-machine communications.

Manuscript received July 6, 1967; revised December 6, 1967, and February 7, 1968. This paper is based on lectures on pattern recognition given at the University of Nebraska Computing Center and also used in a course at the Département d'Informatique, Université de Montréal. Much of the material in the first two sections was presented at the 1967 Symposium on Speech Communication and Processing, and published in *IEEE Trans. Audio and Electroacoustics*, vol. AU-16, June 1968, under the title "Classification Algorithms in Pattern Recognition."

The author is with the Thomas J. Watson Research Center, IBM Corporation, Yorktown Heights, N. Y. 10598. He is currently on leave at the University of Montreal, Montreal 3, Canada.

Other areas where much effort is being expended to develop viable systems include aerial and microphotograph processing, particle tracking in cloud, bubble, and spark chambers, seismic signal analysis for both geophysical exploration and explosion monitoring, electrocardiogram, electroencephalogram, and single fiber recording in medicine and physiology, fingerprint analysis, and many others. A few individual approaches to some of these challenging problems are outlined in Section VI.

II. Conventional Classification Methods

The primary goal in designing a pattern classifier is to have it perform well (achieve a high recognition rate) on new data. When the training data are representative of the test data, and when a very large number of training patterns are available, it is usually argued that it is sufficient to design the classifier to perform adequately on the training set. In practice, the training set is always too small, and extrapolation of the performance figures to new data is hazardous [Allais '64, Glanz '65].

With some a priori information about the nature of the underlying probability distributions, it is indeed possible to predict from a limited training set the performance on the test set. In real problems, however, even the probability model must be inferred from the training set. In the face of this dilemma, the reader must be cautioned that it is possible to *overdesign* the classifier by tailoring it too closely to the training set at the expense of performance on the test set. Matching the design method to the number of samples available is not easy, but a simple rule of thumb is that the more complicated the method, the more samples are required.

In the following discussion, the ith pattern will be treated as a column vector $\bar{x}_i$, and $\bar{x}_i'$ is the corresponding row vector. The components x_{ij} of $\bar{x}_i$ denote individual observations: the energy around 300 Hz in the first 100 ms of an utterance, whether the lower left-hand corner of a character is black or white, the temperature of a hospital patient, the location of the peak in an electrocardiogram. In some problems the choice of observations is critical. In others, a natural set of coordinates, such as the gray levels in the matrix representation of a photograph, exists.

One important distinction between pattern recognition and other related disciplines, such as automatic control theory, switching theory, and statistical hypothesis testing, is the high dimensionality of the vectors $\bar{x}_i$. Were it not for the fact that $\bar{x}_i$ typically runs to hundreds of components, with hundreds, thousands, or even millions (as in the bank font problem) of samples, we could undertake calculations far more sophisticated than those discussed in this review.

In the beginning we will confine our attention to two-class problems. In principle, any multiclass problem can be treated as a number of two-class problems involving the separation of each class from the remainder of the universe, but this does not, as a rule, lead to the most economical solution. Several heuristics which bear on the problem of the optimal assignment of a given number of *discriminants* to more than two classes can be found in [Braverman '62] and in [Kiessling '65]. In the following, the class will always be denoted by a superscript.

All of the categorization methods where the components of the pattern vector are not limited to binary numbers will be illustrated by means of the same two-dimensional example. In this example there are eight patterns in the training set, four from each class:

$$\text{class } C^1$$

$$\bar{x}_1 \quad \bar{x}_2 \quad \bar{x}_3 \quad \bar{x}_4$$

$$\begin{pmatrix} 0 \\ 2+2\sqrt{2} \end{pmatrix} \begin{pmatrix} 0 \\ 2-2\sqrt{2} \end{pmatrix} \begin{pmatrix} \sqrt{6} \\ 2 \end{pmatrix} \begin{pmatrix} -\sqrt{6} \\ 2 \end{pmatrix}$$

$$\text{class } C^2$$

$$\bar{x}_5 \quad \bar{x}_6 \quad \bar{x}_7 \quad \bar{x}_8$$

$$\begin{pmatrix} 4 \\ 2\sqrt{2} \end{pmatrix} \begin{pmatrix} 4 \\ -2\sqrt{2} \end{pmatrix} \begin{pmatrix} 4+\sqrt{2} \\ 0 \end{pmatrix} \begin{pmatrix} 4-\sqrt{2} \\ 0 \end{pmatrix}.$$

In order to test the various methods, it is assumed that these samples originate from multivariate *normal* (or *Gaussian*) populations with means $\bar{\mu}^k$ equal to the sample means, and covariance matrices A^k equal to the sample covariance matrices:

$$\bar{\mu}^1 = \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \quad \bar{\mu}^2 = \begin{pmatrix} 4 \\ 0 \end{pmatrix}, \quad A^1 = \begin{pmatrix} 3 & 0 \\ 0 & 4 \end{pmatrix}, \quad A^2 = \begin{pmatrix} 1 & 0 \\ 0 & 4 \end{pmatrix}.$$

The samples, and the elliptical equiprobability contours of the normal density functions

$$p^k(\bar{x}) = (2\pi)^{-n/2} (\det A^k)^{-1/2} \exp\left[-\tfrac{1}{2}(\bar{x} - \bar{\mu}^k)' A^{k-1} (\bar{x} - \bar{\mu}^k)\right]$$

with the above parameters, are shown in Fig. 2.

Of course, in a real problem one could hardly expect the distribution of the test samples to correspond so precisely to the distribution specified by the sample means and covariance matrices, especially with such a small training sample. For this reason, the computed performance figures for the different methods should be treated with extreme caution; the numerical results are really intended only as check points for readers interested in gaining first-hand experience with some of the formulas.

Linear Classification

A *linear categorizer* assigns an unknown pattern $\bar{x}_i$ to class C^1 if $\bar{x}_i' \cdot \bar{w} \geq \theta$, and to class C^2 otherwise. The coefficients w_j of $\bar{w}$ are proportional to the components of a vector (through the origin) onto which the patterns are projected. In the two-dimensional example in Fig. 1, all the points which are to the left of the dotted straight line perpendicular to the vector $(w_1/\sqrt{w_1^2 + w_2^2}, w_2/\sqrt{w_1^2 + w_2^2})$ at a distance $\theta/\sqrt{w_1^2 + w_2^2}$ from the origin are assigned to class C^1.

When $\bar{x}_i$ has more than two components, it is still projected onto the vector $\bar{w}$, but now a *hyperplane*, rather than a line, separates the classes. $\bar{w}$ is traditionally referred to as the *weight vector*, because its components represent the relative importance of each observation in deciding the class assignment.

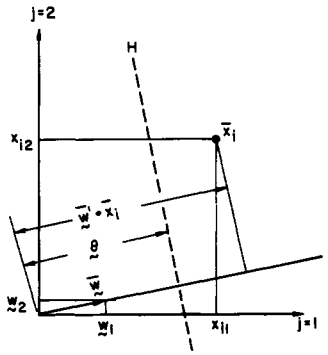


Fig. 1. A linear categorizer. The normalized weight vector $\bar{w}$ and the normalized threshold θ define a two-class linear categorizer. The pattern $\bar{x}_i$ is assigned to class C^1 because its projection $\bar{w}'\bar{x}_i$ on the normalized weight vector is greater than θ. Any pattern on the same side of hyperplane H as $\bar{x}_i$ would be assigned to C^1.

The *weights* w_j and the *threshold* θ may be multiplied by any constant without a change in the resulting classification. It is customary to normalize the weight vector to unit length.

In the two-dimensional example we propose to discuss, the set is assumed to consist of an infinite number of patterns with known Gaussian distributions. The error for any linear categorizer would be computed by projecting these distributions onto the weight vector. The projected one-dimensional distributions are also Gaussian, with means $\bar{w}'\bar{\mu}^k$ and variances $\bar{w}'A^k\bar{w}$. The error rate can be readily found from a table of cumulative Gaussian probabilities:

$$P(\text{error}) = \tfrac{1}{2}\phi\left[\frac{\theta - \bar{w}'\bar{\mu}^1}{(\bar{w}'A^1\bar{w})^{1/2}}\right] + \tfrac{1}{2} - \tfrac{1}{2}\phi\left[\frac{\theta - \bar{w}'\bar{\mu}^2}{(\bar{w}'A^2\bar{w})^{1/2}}\right]$$

where both classes are equally probable and

$$\phi(y) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{y} e^{-y^2/2}dy.$$

The weight vectors, hyperplanes, and other decision boundaries obtained by the various categorization procedures are shown in Fig. 2. The computed error rates rates appear in Table I.

Piecewise linear categorizers simply contain more than one weight vector to characterize a class pair. Boundaries of arbitrary complexity may be approximated by the resultant profusion of hyperplanes [Duda '66, Rosen '66].

Correlation

The simplest method of computing the parameters of the linear categorizer is to let $\bar{w} = \bar{c}^1 - \bar{c}^2$, where $\bar{c}^1$ and $\bar{c}^2$ represent "typical" members of the two classes. Customarily, $\bar{c}^1$ is the set equal to $\bar{\mu}^1$, the centroid of class C^1, and $\bar{c}^2 = \bar{\mu}^2$, the centroid of C^2. The categorization operation is then equivalent to a component-by-component correlation of the unknown sample with the sample means of the classes. Thus

$$\bar{w} = \bar{\mu}^1 - \bar{\mu}^2$$

and

$$\theta = \tfrac{1}{2}(\bar{\mu}^1 - \bar{\mu}^2)'(\bar{\mu}^1 + \bar{\mu}^2). \tag{1}$$

The resulting hyperplane is shown as H_1 in Fig. 2. When it is felt that the magnitude of the feature vectors matters less than their orientation, $\bar{w}$ is set equal to $(\bar{\mu}^1/|\bar{\mu}^1|) - (\bar{\mu}^2/|\bar{\mu}^2|)$, with a threshold of 0. This is hyperplane H_2 in Fig. 2. Such a decision procedure would be useful, for example, in classifying sustained vowel sounds where the overall intensity depends only on the distance from the microphone, and the sound is fully characterized by the relative intensities of the various frequencies.

The correlation method is often used on binary data, where it is sometimes referred to as mask or template matching. For computational efficiency the number of mismatching, rather than matching, bit positions is usually computed, and the calculation is truncated when this reaches a preset threshold. The use of additive and multiplicative constants in normalization sometimes results in startling changes in performance.

For continuous variables correlation with the sample means is readily implemented with the *Lernmatrix*. This ingenious formalism allows the concatenation of several correlators, as well as "bidirectional" classification [Steinbuch '63].

Theoretically, the correlation methods can be shown to be optimal only under certain very restrictive symmetry conditions on the distributions.

Maximum Likelihood

The application of the maximum likelihood principle to pattern classification makes use of Bayes' formula for conditional probabilities to show that in order to determine the greater of $P[C^1|\bar{x}_i]$ and $P[C^2|\bar{x}_i]$ it is sufficient to compare $P[\bar{x}_i|C^1]$ and $P[\bar{x}_i|C^2]$ (for equal a priori probabilities on the classes). This means that we must ask which class was *most likely* to produce the observed sample vector.

The determination of the conditional probability of the pattern vector, given the class, leads to a linear expression in the components of the pattern vector under several assumptions.

When the components x_{ij} are *statistically independent* of one another,

$$P[\bar{x}_i|C^k] = \prod_j P[x_{ij}|C^k].$$

Geometrically this condition corresponds to having the principal axes of the two distributions parallel to the coordinate axes. If, in addition, the x_{ij}'s are *binary*, it can be shown [Minsky '61] that

$$P[C^1|\bar{x}_i] \geq P[C^2|\bar{x}_i] \text{ if and only if } \bar{x}_i' \cdot \bar{w} \geq \theta,$$

where

$$w_j = \ln \frac{P[x_{ij} = 1|C^1]P[x_{ij} = 0|C^2]}{P[x_{ij} = 1|C^2]P[x_{ij} = 0|C^1]}$$

and

$$\theta = \sum_j \ln \frac{P[x_{ij} = 0|C^2]}{P[x_{ij} = 0|C^1]}. \tag{2}$$

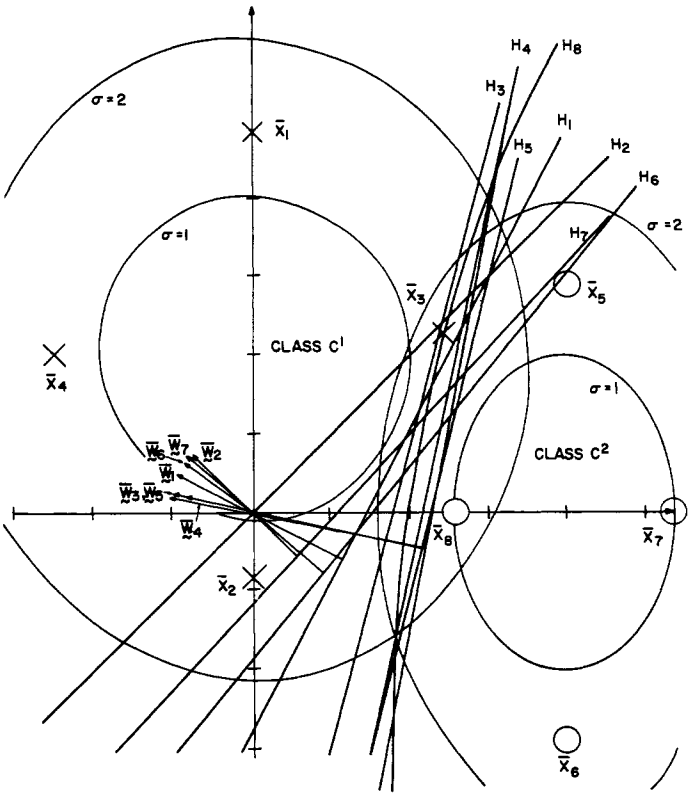


Fig. 2. Common types of linear categorizers. × and ○ indicate the training samples in classes C^1 and C^2, respectively. The ellipses are the equiprobability contours on the postulated distributions in the test data. The subscripts associated with the hyperplanes and weight vectors pertain to the following categorizers: 1) distance to means; 2) correlation; 3) approximate maximum likelihood: 4) Anderson-Bahadur; 5) discriminant analysis; 6) approximate discriminant analysis; 7) trainable machine; 8) optimal quadratic boundary.

Despite the fact that the independence assumption is very seldom satisfied in practice, this decision method has been widely used in cases where the high dimensionality of the pattern vectors precludes more complicated calculations. Let us compute the parameters and the error rate using this scheme in another two-dimensional example (where the patterns are restricted to binary components):

Class	Pattern	Number of Patterns in Training Sample
C^1	(0, 0)	60
	(1, 1)	40
C^2	(1, 0)	30
	(0, 1)	70

The probabilities needed to estimate the components of the weight vector and the threshold are readily calculated:

$$P[x_{i1} = 1|C^1] = 0.4, \quad P[x_{i2} = 1|C^1] = 0.4,$$
$$P[x_{i1} = 1|C^2] = 0.3, \quad P[x_{i2} = 1|C^2] = 0.7.$$

The weight vector and the resulting hyperplane calculated from (2) are shown in Fig. 3. Since the independence assumption is clearly violated, we need not be surprised that the error rate obtained with this plane on the training sample, 0.35, is higher than that obtained with other planes. The plane shown in dotted lines in Fig. 3, for example, yields only 15 percent errors. With a nonlinear scheme we could separate the classes without error, since there is no overlap between the distributions.

With a finite number of samples in the training set it may happen that a numerator or a denominator in (2) vanishes.

TABLE I

No. in Fig. 2	Method	Parameters				% Error Rate on Test Data		
		Unnormalized		Normalized		on C^1	on C^2	Average
		$\bar{w}$	θ	$\tilde{\bar{w}}$	$\tilde{\theta}$			
1	Distance from means	−4.00 2.00	−6.00	−0.89 0.45	−1.34	11.5	3.8	7.7
2	Correlation	−1.00 1.00	0.00	−0.71 0.71	0.00	22.7	3.7	13.2
3	Approximate maximum likelihood	−2.00 0.50	−3.50	−0.96 0.24	−1.70	10.7	2.2	6.5
4	Anderson-Bahadur	−2.27 0.50	−5.20	−0.98 0.21	−2.26	5.9	5.9	5.9
5	Discriminant analysis	−4.00 1.00	−9.15	−0.97 0.24	−2.21	6.3	6.3	6.3
6	Approximate discriminant analysis	−1.28 1.00	−3.10	−0.79 0.61	−1.21	4.4	19.8	12.1
7	Trainable machine	−3.89 3.85	−4.00	−0.72 0.70	−0.73	12.5	9.2	10.8
8	Optimum quadratic boundary							~5.6

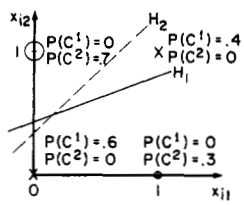


Fig. 3. Categorizers for binary patterns. Class 1 is indicated by circles, class 2 by crosses. The size of the symbols is proportional to the postulated probability density distribution. H_1 is the hyperplane calculated from (2); it is not as good as H_2, which could be obtained by inspection because of the low dimensionality of the problem. The unnormalized weight vector corresponding to H_1 is (0.441, −1.25), with a threshold of −0.41.

To avoid this problem there is some theoretical justification for estimating $P[x_{ij}=1|C^k]$ by means of $(M_j^k+1)/(N^k+2)$, instead of M_j^k/N^k as above, where M_{jk} is the number of samples in class k with the jth component equal to one, and N^k is the total number of samples in class k [Good '65].

Another instance where the maximum likelihood classifier is linear is in the case of *normal* distributions with *identical covariance* matrices. The form of the distributions is again

$$p^k(\bar{x}) = \frac{1}{(2\pi)^{n/2}|A|^{1/2}} \exp\left[-\frac{1}{2}(\bar{x}-\bar{\mu}^k)'A^{-1}(\bar{x}-\bar{\mu}^k)\right],$$

where the elements of the matrix A are the same whichever class k is used to derive them:

$$A = \underset{k}{E}\left[(\bar{x}-\bar{\mu}^k)(\bar{x}-\bar{\mu}^k)'\right]$$

and

$$\bar{\mu}^k = \underset{k}{E}\left[\bar{x}\right].$$

The assumption of equal covariances is reasonable, for example, in digitized photographs, where the adjacent cells are likely to be positively correlated regardless of class because the gray scale seldom contains rapid transitions.

In comparing the ratio of the probability distribution functions to one, the exponents subtract and the second-order terms in the pattern components cancel. In the resulting linear expression,

$$\bar{w} = (\bar{\mu}^1 - \bar{\mu}^2)'A^{-1}$$

and

$$\theta = \tfrac{1}{2}(\bar{\mu}^1 - \bar{\mu}^2)'A^{-1}(\bar{\mu}^1 + \bar{\mu}^2). \tag{3}$$

To apply this method to our example, we shall approximate A by the mean of the covariance matrices for the two classes. Thus

$$A = \begin{pmatrix} 2 & 0 \\ 0 & 4 \end{pmatrix}, \quad \text{and} \quad A^{-1} = \begin{pmatrix} \frac{1}{2} & 0 \\ 0 & \frac{1}{4} \end{pmatrix}.$$

The resulting hyperplane is shown as H_3 in Fig. 2.

If the distributions are spherical, the covariance matrix is proportional to the identity matrix, and the weight vector corresponds to the difference of the centroids, as in simple correlation.

When the number of samples in the training set is small compared to the dimensionality of the pattern vectors, the covariance matrix may be singular. Without the inverse, one must either guess the values of the variance in the missing directions, or confine the solution weight vector to the subspace actually spanned by the training set [Penrose '55, Raviv '65].

Minimax Decision Rule—The Anderson-Bahadur Formula

With *normal distribution* functions characterized by *unequal covariance* matrices, the maximum likelihood boundary is nonlinear. Therefore, the minimax criterion, which equalizes the probabilities of the two kinds of errors (for equal a priori probabilities), is used instead.

The following implicit equation for the weight vector $\bar{w}$ has been derived by Anderson and Bahadur [Anderson '62]:

$$\bar{w} = \left[\frac{(\bar{w}'A^2\bar{w})^{\frac{1}{2}}}{(\bar{w}'A^1\bar{w})^{\frac{1}{2}}+(\bar{w}'A^2\bar{w})^{\frac{1}{2}}}A^1 + \frac{(\bar{w}'A^1\bar{w})^{\frac{1}{2}}}{(\bar{w}'A^1\bar{w})^{\frac{1}{2}}+(\bar{w}'A^2\bar{w})^{\frac{1}{2}}}A^2\right]^{-1}\cdot(\bar{\mu}^1-\bar{\mu}^2) \tag{4}$$

where

μ^k = mean of class k
A^k = covariance matrix of class k.

Equation (4) can be solved with conventional iterative methods using matrix inversion. When the dimensionality is high, it is desirable to avoid matrix inversion with the method of *conjugate gradients*, which is guaranteed to converge in at most a number of steps equal to the dimensionality [Hestenes '52].

The probability of classification errors with the optimum threshold can be computed in terms of the weight vector:

$$P\,(\text{error}) = 1 - \phi\left[\frac{(\bar{w}'A^1\bar{w})^{\frac{1}{2}}(\bar{w}'A^2\bar{w})^{\frac{1}{2}}}{(\bar{w}'A^1\bar{w})^{\frac{1}{2}}+(\bar{w}'A^2\bar{w})^{\frac{1}{2}}}\right]$$

where

$$\phi(y) = \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{y} e^{-\frac{1}{2}y^2}\,dy.$$

The Anderson-Bahadur hyperplane is shown as H_4 in Fig. 2.

Discriminant Analysis

When the form of the probability density functions governing the distribution of the pattern vectors is not known at all, the minimum-error hyperplane cannot be specified analytically. In this case it seems intuitively desirable to find at least the direction in which the projections of the samples of each class fall as far as possible from those of the other class, but the *internal scatter* of each class is minimized. This is the object of *discriminant analysis* [Wilks '60, Peterson '65].

More formally, we wish to maximize

$$\sum_{\substack{\bar{x}_i\in C^1\\ \bar{x}_j\in C^2}} (\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j)'(\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j),$$

subject to the constraint

$$\sum_{\substack{\bar{x}_i \in C^1 \\ \bar{x}_j \in C^1}} (\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j)'(\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j) + \sum_{\substack{\bar{x}_i \in C^2 \\ \bar{x}_j \in C^2}} (\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j)'(\bar{w}'\bar{x}_i - \bar{w}'\bar{x}_j) = \text{constant}.$$

It can be shown, by using Lagrange multipliers, that the vector $\bar{w}$ which fulfills these conditions is the eigenvector associated with the largest eigenvalue λ of

$$(BA^{-1} - \lambda I)\bar{w} = 0 \tag{5}$$

where A is the intraclass sample scatter matrix

$$A = \sum_{\bar{x}_i \in C^1} (\bar{x}_i - \bar{\mu}^1)(\bar{x}_i - \bar{\mu}^1)' + \sum_{\bar{x}_i \in C^2} (\bar{x}_i - \bar{\mu}^2)(\bar{x}_i - \bar{\mu}^2)'$$

and B is the interclass sample scatter matrix

$$B = \sum_{\text{all } i} \left(\bar{x}_i - \frac{\bar{\mu}^1 + \bar{\mu}^2}{2}\right)\left(\bar{x}_i - \frac{\bar{\mu}^1 + \bar{\mu}^2}{2}\right)' - A.$$

The solution vector $\bar{w} = A^{-1}(\bar{\mu}^1 - \bar{\mu}^2)$ is identical to the approximation we used for the maximum likelihood solution in (3), but explicit use of the individual covariance matrices to equalize the two types of errors leads to a better choice of threshold. The hyperplane obtained by determining the values of θ for which $\phi[(\theta - \bar{w}'\bar{\mu}^1)/(\bar{w}'A^1\bar{w})^{\frac{1}{2}}]$ and $\phi[(\bar{w}'\bar{\mu}^1 - \theta)/(\bar{w}'A^2\bar{w})^{\frac{1}{2}}]$ are equal is shown as H_5 in Fig. 2.

Another possible approximation consists of replacing the intraclass scatter matrix A in (5) by the identity matrix. This corresponds to maximizing the scatter of the projected points in the two classes pooled together, and is also equivalent to principal components analysis. This hyperplane, with $\theta = \frac{1}{2}\bar{w}'(\bar{\mu}^1 + \bar{\mu}^2)$, is shown as H_6 in Fig. 2.

Trainable Categorizers

In applying the statistical algorithms of the preceding sections, all of the patterns in the training set were considered simultaneously to compute the weight vector. In a trainable categorizer, however, the patterns are presented one at a time, and the weight vector is changed incrementally throughout the process.

In contradistinction to statistical methods, no formal hypothesis is postulated regarding the distribution of the samples. It is merely assumed that if all of the training samples are correctly classified, few mistakes will be made on the test data. This mode of operation offers some advantage in implementing the algorithm in hardware, but the final error rates achievable by these approaches appear to be substantially the same [Kanal '62, Bryan '63, Konheim '64, Casey '65[2]].

There are many algorithms which guarantee convergence to the optimal weight vector under various conditions [Nilsson '64]. One of the earliest for which a bound on the maximum number of steps required was obtained is the *error correcting* algorithm of perceptron fame [Rosenblatt '57]. Here all the patterns in the training set are presented in sequence; when the training set is exhausted, the patterns are presented again in the same order. Thus, if there are N patterns available, the jth step involves the ith pattern, where $i = j$ modulo N. The weights change according to

$$\begin{aligned} \bar{w}_{j+1} &= \bar{w}_j + \bar{x}_j \quad \text{if } \bar{w}_j\bar{x}_j \le \theta \text{ and } \bar{x}_j \in C^1 \\ &= \bar{w}_j - \bar{x}_j \quad \text{if } \bar{w}_j\bar{x}_j \ge \theta \text{ and } \bar{x}_j \in C^2 \\ &= \bar{w}_j \quad \text{otherwise.} \end{aligned} \tag{6}$$

The weight vector is changed only after a pattern has been misidentified. The initial vector $\bar{w}_0$ may be the null vector, or, better, a coarse approximation such as may be obtained with (1). The threshold may be derived by extending the dimensionality of the pattern space by one, and setting the corresponding component of all the patterns equal to one. The weight associated with this component is the required threshold.

The manner of convergence of the algorithm is shown in the two-dimensional problem in Table II. The final weight vector corresponds to hyperplane H_7 in Fig. 2.

Since 1957, when a proof for the convergence of this algorithm was outlined by Rosenblatt, at least six or seven more or less independent proofs have been advanced. One derivation, and the corresponding bound on the number of steps, is given by Singleton in terms of the *pattern matrix B* [Singleton '62]. The ith colum of B is $\bar{x}_i$ if $\bar{x}_i$ belongs to C^1, and $-\bar{x}_i$ if it belongs to C^2.

The theorem states that if $\exists \bar{w} \ni B^t \cdot \bar{w} > 0$, then

$$\bar{w}_{k+1} = \bar{w}_k \quad \text{for } k \ge \frac{(\bar{w} \cdot \bar{w}) \max_i (B^tB)_{ii}}{[\min_i (B^t\bar{w})_i]^2}. \tag{7}$$

This means that if the categorization problem does have a linear solution, an incremental adaptation procedure will find it in a finite number of steps. The similarity to the achievement of feasible solutions in linear programming has been repeatedly pointed out [Rosen '65, Smith '68].

The upper bounds given in the literature for the number of adjustments needed unfortunately all require that at least one solution to the problem be known before the length of the training sequence can be estimated. In the example in Table II the bound calculated from (7) is 284 steps, whereas convergence is actually reached in 45 steps. This bound is a very sensitive function of the smallest distance of a pattern point from the separating plane used to estimate the bound.

Variations of the theorem deal with the effects of varying the order of presentation of the training patterns, of changing the total amount added to the weights depending on how close the response is to being right, and of using imperfect components [Joseph '60, Kesler '61, Highleyman '62, Hoff '62, Ridgeway '62, Griffin '63, Low '63, Mays '63].

Judicious adjustment of the size of the corrective increment, for example, depending on how close the response is to being right and on the running average of the fraction of misidentified patterns, yields convergence rates many orders of magnitude faster than the original algorithm. It appears customary, however, to demonstrate the power of new methods of solving linear inequalities by comparing their speed to the long disused equal increment algorithm.

Consideration of the positive quadrant of the vector space containing the solution vector $A\bar{w}$ of the system of linear inequalities $Aw > 0$, rather than the space containing $\bar{w}$, leads to an elegant demonstration of a family of convergent

TABLE II
TRAINING A CATEGORIZER

Step	Pattern	$\bar{w}'\bar{x}-\theta$	w_1	w_2	θ	Increment?
1	$\bar{x}_1$	0.00	0.00	4.83	−1.00	
2	$\bar{x}_5$	−14.67	−4.00	2.00	−0.00	*
3	$\bar{x}_2$	− 1.66	−4.00	1.17	−1.00	•
4	$\bar{x}_6$	18.36	−4.00	1.17	−1.00	
5	$\bar{x}_3$	− 6.46	−1.55	3.17	−2.00	*
6	$\bar{x}_7$	5.96	−1.55	3.17	−2.00	
7	$\bar{x}_4$	12.14	−1.55	3.17	−2.00	
8	$\bar{x}_8$	2.01	−1.55	3.17	−2.00	
9	$\bar{x}_1$	17.31	−1.55	3.17	−2.00	
10	$\bar{x}_5$	− 4.77	−5.55	0.34	−1.00	•
11	$\bar{x}_2$	0.72	−5.55	0.34	−1.00	
12	$\bar{x}_6$	22.16	−5.55	0.34	−1.00	
13	$\bar{x}_3$	−11.92	−3.10	2.34	−2.00	•
14	$\bar{x}_7$	13.93	−3.10	2.34	−2.00	
15	$\bar{x}_4$	14.28	−3.10	2.34	−2.00	
16	$\bar{x}_8$	6.03	−3.10	2.34	−2.00	
17	$\bar{x}_1$	13.30	−3.10	2.34	−2.00	
18	$\bar{x}_5$	3.78	−3.10	2.34	−2.00	
19	$\bar{x}_2$	0.06	−3.10	2.34	−2.00	
20	$\bar{x}_6$	17.02	−3.10	2.34	−2.00	
21	$\bar{x}_3$	− 0.92	−0.65	4.34	−3.00	*
22	$\bar{x}_7$	0.34	−0.65	4.34	−3.00	
23	$\bar{x}_4$	13.27	−0.65	4.34	−3.00	
24	$\bar{x}_8$	− 1.32	−3.24	4.34	−2.00	*
25	$\bar{x}_1$	22.97	−3.24	4.34	−2.00	
26	$\bar{x}_5$	− 1.32	−7.24	1.51	−1.00	*
27	$\bar{x}_2$	− 0.25	−7.24	0.68	−2.00	*
28	$\bar{x}_6$	28.88	−7.24	0.68	−2.00	
29	$\bar{x}_3$	−14.38	−4.79	2.68	−3.00	*
30	$\bar{x}_7$	21.62	−4.79	2.68	−3.00	
31	$\bar{x}_4$	20.10	−4.79	2.68	−3.00	
32	$\bar{x}_8$	9.41	−4.79	2.68	−3.00	
33	$\bar{x}_1$	15.94	−4.79	2.68	−3.00	
34	$\bar{x}_5$	8.58	−4.79	2.68	−3.00	
35	$\bar{x}_2$	0.78	−4.79	2.68	−3.00	
36	$\bar{x}_6$	23.74	−4.79	2.68	−3.00	
37	$\bar{x}_3$	− 3.38	−2.34	4.68	−4.00	*
38	$\bar{x}_7$	8.08	−2.34	4.68	−4.00	
39	$\bar{x}_4$	19.09	−2.34	4.68	−4.00	
40	$\bar{x}_8$	2.06	−2.34	4.68	−4.00	
41	$\bar{x}_1$	26.60	−2.34	4.68	−4.00	
42	$\bar{x}_5$	− 7.88	−6.34	1.85	−3.00	*
43	$\bar{x}_2$	1.46	−6.34	1.85	−3.00	
44	$\bar{x}_6$	27.59	−6.34	1.85	−3.00	
45	$\bar{x}_3$	− 8.83	−3.89	3.85	−4.00	*
46	$\bar{x}_7$	15.99	−3.89	3.85	−4.00	
47	$\bar{x}_4$	21.23	−3.89	3.85	−4.00	
48	$\bar{x}_8$	6.08	−3.89	3.85	−4.00	
49	$\bar{x}_1$	22.60	−3.89	3.85	−4.00	
50	$\bar{x}_5$	0.66	−3.89	3.85	−4.00	
51	$\bar{x}_2$	0.80	−3.89	3.85	−4.00	
52	$\bar{x}_6$	22.46	−3.89	3.85	−4.00	
53	$\bar{x}_3$	2.17	−3.89	3.85	−4.00	
54	$\bar{x}_7$	15.99	−3.89	3.85	−4.00	
55	$\bar{x}_4$	21.23	−3.89	3.85	−4.00	
56	$\bar{x}_8$	6.08	−3.89	3.85	−4.00	

class C^1:

$$\bar{x}_1 = \begin{pmatrix} 0.00 \\ 4.83 \end{pmatrix}$$

$$\bar{x}_2 = \begin{pmatrix} 0.00 \\ -0.83 \end{pmatrix}$$

$$\bar{x}_3 = \begin{pmatrix} 2.45 \\ 2.00 \end{pmatrix}$$

$$\bar{x}_4 = \begin{pmatrix} -2.45 \\ 2.00 \end{pmatrix}$$

class C^2:

$$-\bar{x}_5 = \begin{pmatrix} 4.00 \\ 2.83 \end{pmatrix}$$

$$-\bar{x}_6 = \begin{pmatrix} 4.00 \\ -2.83 \end{pmatrix}$$

$$-\bar{x}_7 = \begin{pmatrix} 5.41 \\ 0.00 \end{pmatrix}$$

$$-\bar{x}_8 = \begin{pmatrix} 2.59 \\ 0.00 \end{pmatrix}$$

training algorithms [Ho '65]. For a large number of pattern vectors, however, computational difficulties may arise in the determination of the required $m \times m$ matrix $AA^{\&}$, where m is the number of patterns and & denotes the generalized inverse.

Joseph has proved that the fundamental convergence theorem holds even if the number of levels in each adaptive link is finite, i.e., if the storage elements are saturable. Low has a similar demonstration for the case of nonuniform adaptation.

Perturbations in the final values of individual weights introduce errors. Hoff has shown that the expected probability of errors, on patterns with a uniform distribution of ones and zeros, is roughly

$$p(\varepsilon) = \frac{3}{4} \frac{\delta}{|\bar{w}|} \sqrt{n}$$

where δ is the average drift in the weights, $\bar{w}$ is the solution weight vector (before drifting), and n is the number of weights.

The convergence theorem, with all its variants and corollaries, applies only if a solution exists. It is not difficult to show that solutions exist if and only if there are no linear dependencies between input patterns, considered as vectors, in opposite classes. In other words, conflicts of the type seen in the example in Fig. 3 must not arise.

It is no trivial matter, however, to look at several thousand 100- or 500-dimensional vectors and spot the linear dependencies. A number of procedures, some of which take advantage of the statistical distribution of ones and zeros in the input vectors, have been devised to carry out this operation, but the most common method for finding out if a problem is linearly separable is to simulate the linear categorizer on a computer and try to teach it the required classification [Singleton '62, Gaston '63, Sheng '64]. If, after many presentations of the pattern, it has not been learned, then it is assumed that unwanted linear dependencies do occur. Several adapters have noticed the oscillatory behavior of the weight vector when presented with an insoluble task; this symptom of frustration provides a valuable clue as to when to stop training [Ridgeway '62, Kesler '63, Efron '63].

It would be comforting to know that, if the problem is not completely solvable, the weights converge to the values guaranteeing a minimum number of mistakes among the training samples. This, however, is not necessarily the case; the algorithm can be stranded on a local optimum.

Chow points out that the assumptions leading to the procedures specified by (2) and (6), statistical independence and linear separability, are mutually contradictory for binary patterns, except in trivial instances [Chow '65]. To circumvent this difficulty, several gradient methods, which cover the gamut between the "little at a time" and the "all at once" approaches, have been developed [Duda '64, Groner '64, Koford '64].

Nonlinear Categorizers

With the exception of a few special distributions, very little is known about approximating the optimum nonlinear boundaries in classification problems involving pattern vectors with numerical (as opposed to binary) correlated components. For *Gaussian* distributions the optimal separating surface is easily shown to be a hyperquadric, specified by the following equations [Cooper '64]:

$$(\bar{x} - \mu^1)' A^{1^{-1}} (\bar{x} - \bar{\mu}^1) + \log \det A^1 = (\bar{x} - \bar{\mu}^2)' A^{2^{-1}} (\bar{x} - \bar{\mu}^2) + \log \det A^2 \quad (8)$$

where A^k is the covariance matrix of class k and $\bar{\mu}^k$ is the mean vector of class k. This boundary is sketched as H_8 in Fig. 2 for the distributions used to illustrate the linear methods.

In addition to Gaussian distributions, the hyperquadric is optimal for *Pearson distributions type II and type VII*; these cases have also been fully analyzed by Cooper. The equations for the separating surface for a pair of parametric distributions can, of course, always be obtained by setting the difference of the density functions equal to zero, but since in a practical situation the difficulty usually lies in estimating the parameters from the samples, this is really begging the question.

In an n-dimensional *binary* classification task, every vertex of the n-dimensional hypercube represents a possible pattern. Thus the general solution requires 2^n values for complete specification, as opposed to the n values for the linear separation. In practice, even the n^2 values required by considering all of the components *pairwise* without additional simplifying assumptions represent too much of a burden for implementation or simulation.

The probability distributions we must compare in order to decide the class assignment can be written as product expansions. In the following expressions the single subscript refers to the components of the pattern vector:

$$P(\bar{x}|C^k) = P(x_1|C^k)P(x_2|x_1, C^k) \cdots \cdot P(x_j|x_{j-1}, x_{j-2}, \cdots, x_1, C^k) \cdots .$$

Each variable is conditioned upon all the variables appearing in the preceding factors. The product can be rewritten as a sum of weighted terms by taking logarithms and assigning a weight component to every possible sequence of ones and zeros in the partial pattern vector in each term. Thus

$$\ln P(\bar{x}|C^k) = w_1 x_1 + w_2(1 - x_1) + w_3 x_1 x_2 + w_4(1 - x_1)x_2 + w_5 x_1(1 - x_2) + w_6(1 - x_1)(1 - x_2) \cdots .$$

The customary procedure is to neglect all but the second-order terms (with each component conditioned on only one other component), and to select even among these only the most important pairs. In some cases a natural ordering is available. For example, if the pattern is the binary matrix representation of a character, it is reasonable to let each x_{ij} depend only on its "nearest neighbors" [Chow '62].

If, however, the binary components represent the results of arbitrary measurements upon the patterns, then a natural ordering is not available, and the most important pairs must be found by heuristic methods. "Chain" and "tree" representations based on performance criteria are advocated in [Chow '66]. An alternative is to let a trainable machine adapt its weights on inputs representing every pair in all four possible combinations, and select the pairs with the largest weights for ultimate use.

A simple illustration of the effect of correlation among the pattern components is given in Fig. 4. Here two patterns differ from a third by the same number of bits, but in one case the mismatching locations appear to be highly correlated, while in the other they are independently distributed.

The selection of correlated points in a pattern is closely related to the problem of measurement design or feature extraction, and will be discussed further from that point of view in Section V.

Another nonlinear method is the *nearest neighbor decision* (which is not related to the nearest neighbor correlation discussed above). Here each pattern in the test set is assigned to the class of the pattern closest to it (in terms of an arbitrary metric) in the training set. It can be shown that for an infinitely large training sample the error rate obtained

Fig. 4. Correlations in binary patterns. The center pattern, which may be considered the "unknown," differs from each of the outside patterns ("templates") by 9 bit positions. The effect of the correlations among the mismatch bits must thus be taken into account for correct identification. Although this is an artificially constructed example, instances of such neighborhood correlations frequently occur in practice.

with this classifier is at most twice as high as for an optimal (Bayes) classifier [Cover '66]. Of course, even for a finite training sample it may take a considerable amount of storage to recall every training pattern whenever a test sample is presented.

In 1963, Highleyman offered for public use a quantized set of 1800 handprinted characters. Because of the unavailability of any other standard, these data have been used, in spite of their shortcomings, for numerous comparisons of both linear and nonlinear methods of classification. The latest [Munson '68] of the dozen or so published experiments on this material compares the nearest neighbor decision to several others, including human performance.

Sequential Decisions

Rather than look at all of the available information simultaneously, we may proceed sequentially. Then *whether* we look at another point in the pattern (or combination of points), and *which* other point, depends on the outcome of the previous tests. This approach is closely related to *dynamic programming*, and is particularly well suited for implementation on a digital computer.

Fu, Chen, and Chien have examined in detail the economies in computation which may be realized through the application of sequential decision models in pattern recognition [Fu '64, Chien '66]. The strategy is simple: the next measurement chosen is always the one which gives the most information about the class pair with the highest residual probability of error. The interrogation of measurements is halted when the estimated error probability reaches a preset threshold, or when all the measurements in a particular branch of the decision tree are exhausted. In order to apply the theory to practical problems, a great many assumptions must be satisfied. Nevertheless, small-scale experiments show promising results.

A much simpler form of sequential decision, involving only two levels, has been successfully applied to the recognition of both English and Chinese characters [Liu '66, Casey '66].

Potential Functions and Stochastic Approximation

Before leaving the subject of classification based on identified samples, we should discuss the concepts of potential functions and stochastic approximation. These points of view represent theoretical schools of pattern recognition closely related to both the statistical and the trainable machine approaches.

The object in the method of *potential functions* is to find a function $\psi(\bar{x})$, defined on the pattern space X, which, in the cases of nonoverlapping distributions, is positive for all patterns $\bar{x}$ in C^1, negative for $\bar{x}$ in C^2, and either, or zero, elsewhere. The key assumption is that there exists at least one such function which is sufficiently "smooth" to be expanded in a finite number (m) of terms of some "ordinary" orthonormal system of functions:

$$\psi(\bar{x}) = \sum_{i=1}^{m} w_i \phi_i(\bar{x}).$$

The system ϕ is to be specified ahead of time. If the function ψ is sufficiently well-behaved then it does not matter whether trigonometric, Hermite, Lagrange, or other "ordinary" functions are used, though in general m will depend on the system chosen.

The next step is the transformation of the n-dimensional pattern space X into an m-dimensional "linearization" space Z. The new coordinates of a pattern $\bar{x}$ are $z_1, z_2, \cdots, z_m$, where $z_i = \phi_i(\bar{x})$. This transformation maps the separating surface $\psi(\bar{x}) = 0$ into the hyperplane $\sum_{i=1}^{m} w_i \cdot z_i = 0$.

It can now be readily shown that the error-correcting algorithm applies in the linearization space Z, and therefore the coefficients w_i can be obtained after the presentation of a finite number of training patterns. Rather than dwell on the convergence of the weight vector in Z, it is more interesting to observe the corresponding evolution of the "potential function" ψ in X. Here every correction due to an incorrectly identified pattern $\bar{x}_k$ results in the addition (or subtraction) of an incremental potential

$$K(\bar{x}, \bar{x}_k) = \sum_{i=1}^{m} \phi_i(\bar{x}) \phi_i(\bar{x}_k)$$

to the existing potential.

The function $K(\bar{x}, \bar{x}_k)$ has a maximum at $\bar{x} = \bar{x}_k$. If the separating function ψ is highly convoluted in the original space X, and therefore m is large, then the K's will be positive or negative spikes centered at the incorrectly identified patterns. If the separating function is very smooth, the K's will be broad hummocks affecting the potential in a wide area around the misidentified patterns.

In point of fact, we can dispense altogether with the linearizing process, and simply guess at an appropriate incremental potential function such as

$$e^{-\alpha(\bar{x}-\bar{y})'(\bar{x}-\bar{y})} \quad \text{or} \quad \frac{\sin \alpha |\bar{x} - \bar{y}|}{|\bar{x} - \bar{y}|}.$$

This is particularly advantageous when there are few patterns in the training set; for each test pattern the potential is quickly computed as the sum of terms due to the patterns misidentified during the training sequence. When the number of training patterns is large compared to the dimensionalities of X and Z, it is of course more advantageous to store only the functions ϕ_i and the coefficients w_i.

The framework of stochastic approximation provides a convenient means of formalizing some of these notions. Stochastic approximation is the statistical equivalent of hill-

climbing. There is a function to be approximated, an approximating function, and some measure of the difference between these which is to be minimized. The only information available to construct the approximating function consists of a sequence of noisy observations of the value of the function to be approximated (in general, the *regression function*) at randomly selected points.

For our purposes the sequence of randomly selected points consists of the patterns in the training set. Each $\bar{x}$ is selected with probability $p(\bar{x})$. The function to be approximated may be taken as $f(\bar{x})=P(C^1|\bar{x})-P(C^2|\bar{x})$, which is positive wherever patterns belonging to C^1 predominate and negative in the "region of influence" of C^2. The observations are the sequence $y(\bar{x}_k)$, which take on the value $+1$ or -1 depending on whether $\bar{x}_k$ is in C^1 or C^2. Of course, these observations are noisy only in the case of overlapping distributions, since otherwise $f(\bar{x})$ takes on no intermediate values between $+1$ and -1. The approximating function is a sum of the form $\sum_{i=1}^{m} w_i\phi_i$, where the ϕ_i's are given a priori. The measure of deviation to be minimized is the mean square error

$$\int_X \left[f(\bar{x}) - \sum_{i=1}^{m} w_i\phi_i(\bar{x}) \right]^2 p(\bar{x})d\bar{x}.$$

Under these circumstances, and provided that all the functions meet certain normative conditions, the theory of stochastic approximation assures us that the sequence of estimates

$$w_i(k+1) = w_i(k) + \rho(k)\phi_i(x_k)\left[y(x_k) - \sum_{i=1}^{m} w_i(k)\phi_i(x_k) \right]$$

will converge to the optimum w_i's. The $\rho(k)$ are positive scalars decreasing with increasing k. These scalars, which correspond to the step size in gradient methods, must decrease sufficiently rapidly to avoid oscillations but not fast enough to lead to trapped states. More rigorously,

$$\sum_{k=1}^{\infty} \rho(k) = \infty, \quad \text{and} \quad \sum_{k=1}^{\infty} \rho^2(k) < \infty.$$

If the distributions do not overlap, and if the mean square deviation can be reduced to zero, then stochastic approximation has not added anything new to our knowledge since we have already shown that the separating surface can be determined through the error-correcting algorithm operating in the linearization space. The full power of this theory becomes apparent only when the patterns in the training set cannot be separated according to class either because the distributions overlap or because the ϕ_i's are inadequate to reproduce the separating surface exactly.

In spite of the attractive features of the construct just described, certain practical difficulties have so far prevented widespread application of the theory to actual pattern recognition systems. The classical orthonormal systems of physics do not provide very good "features"; the determination of satisfactory alternatives is really the major part of any recognition problem. Furthermore, approximating a distribution function in the mean square (or other convex function) sense is not always salutary since to minimize the recognition error rate a close approximation is needed above all near the class boundaries.

The method of potential functions is developed in [Aizerman '64[1], '64[2]]. These papers also contain references to earlier and less systematic treatments of this approach; in particular, the genealogy is traced back to Rosenblatt's 1957 simple perceptron. A direct application of the potential function (without the linearization space) is described in [Bashkirov '64]. The relation to stochastic approximation is traced in [Aizerman '65], [Tsypkin '66], [Blaydon '66], and [Kashyap '66]. Speeds of convergence are discussed in [Kashyap '66] and in [Braverman '66[1]]. Possible extensions to unidentified pattern sets (see the following sections) are considered in [Braverman '66[2]] and in [Dorofeyuk '66].

III. Tracking Schemes and Unsupervised Learning

It is clear that when a large training sample truly representative of the expected operating conditions is available, one cannot do better than to train or design a categorizer to perform as well as possible on this training or design sample. If, however, the training set is small, or if one has reason to believe that the composition of the test patterns is likely to undergo systematic changes, then it may be useful to resort to the mode of operation known as tracking, nonsupervised learning, self- or error-correcting behavior, or adaptation.

In practice, training sets are often too small because of the difficulty of associating the correct identities with the samples. In character recognition, for example, one must choose between using synthetic documents all containing the characters in the same order, or real documents, which must be keypunched as well as scanned. If the scanner misses a single very faint character, subsequent characters in the training sample will be misidentified, unless the whole operation is closely monitored.

Another instance of the scarcity of training samples occurs in on-line speech recognition, where each speaker is required, before using the system to enter data, to repeat a string of words in a sample vocabulary. It is evidently advantageous to keep this interlude to a minimum.

Finally, there are many applications where shifts in the nature of the pattern vectors during operation can be readily distinguished from the distortion introduced by the more uncorrelated effects of noise. People's voices change more from day to day than in a given hour, common carrier telephone channels vary their characteristics from call to call but remain relatively constant during a transmission, and a typewritten page usually contains material in only a single typeface (unknown a priori) among the 300 or so currently marketed by United States typewriter manufacturers.

In such cases it is often possible to take advantage of the "local consistency" in the data to improve the average performance. Theory, unfortunately, helps us even less in this endeavor than in the design of "fixed" machines; we shall be

forced more and more to resort to empirical proofs of the usefulness of various algorithms.

An early (1957) example of simple adaptive behavior is supplied by MAUDE, a machine for the automatic recognition of hand-sent Morse code [Gold '58, Selfridge '60]. Only one variable, time, is measured, and there are only three classes of spaces and two classes of marks, but additional structure is introduced by the statistical dependence of successive symbols (marks or spaces). The problem is that some operators' dots are longer than other operators' dashes, and even with the same operator the duration of the symbols tends to shift through the course of the transmission.

MAUDE begins the analysis of a new message with a set of fixed rules such as "the shortest of six successive spaces is almost always a symbol space." On the basis of these rules the first few symbols are identified, and thresholds are established to recognize subsequent symbols by their duration. A running average is kept of the duration of the symbols assigned to the various classes, and the thresholds are continuously readjusted to improve the separation between the classes. The system also makes use of the constraints imposed by the permissible symbol sequences in the Morse alphabet, but higher-order context is not exploited for error correction or detection.

These methods have been applied in a very sophisticated manner to digital data transmission over common carrier lines [Lucky '65, '66]. The principal limitation on the rate of transmission of digital data on voice telephone channels is *intersymbol interference*. In this condition the tails of adjacent pulses mask the pulse being detected. The *adaptive equalizer*, which is a technique of time domain filtering, keeps the distortion of the pulses by the channel to a minimum by computing the tendency to commit a decoding error before an error is actually committed. In addition to the systematic distortion of the waveforms, the equalizer must also cope with random bursts of noise.

During the recent upsurge of interest in *unsupervised learning*, the convergence properties of a number of algorithms operating on a sequence of unidentified samples have been theoretically studied [Cooper '64, Fralick '65, Scudder '65, Patrick '66, Braverman '66[2]]. This problem is closely related to the decomposition of mixture distribution functions, with the added difficulty that an "optimal" solution is desired at each step, rather than only after all the samples have been presented. Successful solutions require restrictive assumptions such as that the distributions of the two classes differ only by a translation, or that the distributions belong to a specific family.

Some of these algorithms have been programmed for a few simple cases (usually one-dimensional Gaussian distributions), but much work remains to be done before they can be applied to "practical" recognition problems. The difficulties inherent in the various approaches have been thoroughly analyzed in a survey paper on unsupervised learning [Spragins '66].

It is clear that information other than the identities of training samples may be used to improve the performance of a categorizer. In addition to the examples we have seen already, this information may take the form of *context* between successive patterns [Raviv '67, Uesaka '67], some long-term or overall success criterion [Widrow '64], statistical confidence in the decision of the categorizer [Ide '66], or the degree of resemblance between the distributions of the unknown patterns and some prior identified training set [Sebestyen '62].

We shall now look at two intuitive, definitely nonoptimal methods of self-adaptation which have given promising results on reasonably realistic pattern sets. Both of these methods are based on the expectation that the initial parameters of the categorizer already permit fairly accurate classification, and that only slight adjustments are needed for further improvements.

An Adaptive Algorithm for Batched Data

When the data are divided into sets exhibiting internal similarity but with differences from set to set, the method illustrated in Fig. 5 may be useful. The original weight vector $\bar{w}_0$, and the corresponding hyperplane H_0, are calculated from an identified training set by one of the formulas given in Section II. For concreteness, (1) will be used, so that H_0 is perpendicular to the line joining the centroids of the two distributions.

The first of the unknown sets, whose distribution may differ considerably from the training set, is now classified by H_0, as shown in Fig. 5. The centroid of the patterns assigned to class C^1 is $\bar{\mu}_1^1$, while the centroid of the patterns assigned to class C^2 is $\bar{\mu}_1^2$.

A new weight vector $\bar{w}_1 = \bar{\mu}_1^1 - \bar{\mu}_1^2$ is now calculated and all the patterns are classified again. The process is iterated until there are no further changes in class membership. More formally:

$$\begin{aligned} \bar{\mu}_j^1 &= \frac{1}{N'} \sum_{i \in \Omega} \bar{x}_i \\ \bar{\mu}_j^2 &= \frac{1}{N - N'} \sum_{i \notin \Omega} \bar{x}_i \\ \bar{w}_j &= \bar{\mu}_j^1 - \bar{\mu}_j^2 \\ \theta_j &= \tfrac{1}{2}|\bar{w}_j| \end{aligned} \tag{9}$$

where $i \in \Omega$ iff $\bar{w}_{j-1} \cdot \bar{x}_i \geq \theta_{j-1}$, N' is the number of integers in Ω, and N is the total number of samples in the set being processed.

On the next batch of data, which may bear little resemblance to the last set, $\bar{w}_0$ is used again for the first pass. The new weight vector is then developed completely independently of the previous batches.

While it is easy to show that the process will not oscillate, the general conditions under which it will converge to the correct classification are not known. Casey has shown that a sufficiency condition for two classes *uniformly* distributed over spheres is that $P^{21}/P^{11} < P^{22}/P^{12}$, where the first superscript indicates the true class, and the second the class assigned by w_0. Thus P^{12} is the fraction of patterns of class C^1 assigned to class C^2 by the initial categorizer.

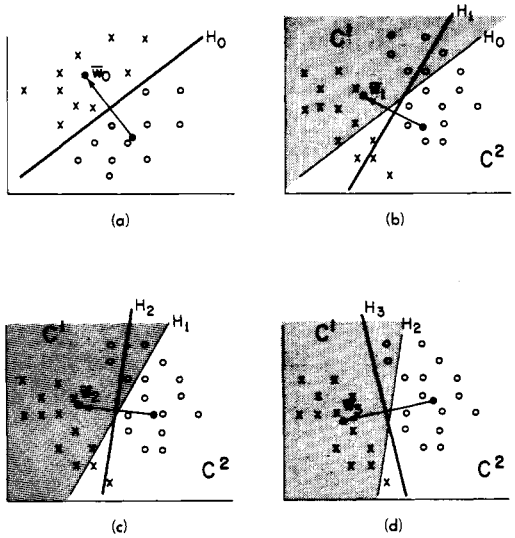


Fig. 5. Adaptive batch processing. (a) Hyperplane H_0 generated by bisecting the difference of the means on design data. (b), (c), and (d) Adaptation on test data. The large dark dots represent the centers of gravity of the patterns in the shaded and unshaded areas. New planes are formed perpendicular to the vector between these points. H_0 results in 10 errors on the test data, H_1 in 6 errors, H_2 in 3 errors, and H_3 in 0 errors. Thus H_4 would be the same as H_3 and the process would come to a halt in the next step.

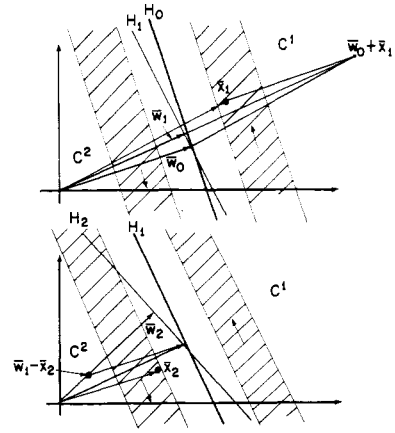


Fig. 6. Continuous tracking. The arrows show a drift in the populations which is being compensated by adapting the hyperplane. The weight vector is changed only when patterns fall in the shaded zones; otherwise the decision is considered either too uncertain, or so secure that it is not necessary to change the plane. In the first step (top), $\bar{x}_1$ is assigned to C^1, so the new weight vector $\bar{w}_1$ is proportional to $\bar{w}_0+\bar{x}_1$. In the next step (bottom), a new pattern $\bar{x}_2$ is assigned to C_2, resulting in a weight vector $\bar{w}_2$ proportional to $\bar{w}_1-\bar{x}_2$. The rotation of the plane from H_0 to H_2 is seen to conform to the needs of the changing distributions.

Some variants of this algorithm have been applied to character recognition. The experimental results are described in [Nagy '66].

An Incrementally Adaptive Algorithm

The error-correcting algorithm described earlier [see (6)] has been adapted to provide continuous updating of the components of the weight vector in classifying slowly changing data.

Only patterns classified with a certain margin of safety are used to adjust the weight vector. When a pattern is either rejected (i.e., it falls in the "no man's land" immediately adjacent to the current hyperplane between the classes), or so strongly assigned to a class that it does not make sense to modify the parameters to change its projection, the weight vector is not altered. When the projection of the pattern falls, however, between the "reject" and the "safe" thresholds, the weight vector is modified by adding the pattern to (if it is in class 1) or subtracting it from (if in class 2) the normalized weight vector. A few cycles of this procedure are shown in Fig. 6.

This algorithm is described by the following equations:

$$\begin{aligned} w_{i+1} &= w_i + s_i x_i \\ \theta_{i+1} &= \theta_i + s_i \\ s_i &= +1 \text{ iff } \theta_i + \varepsilon_1 < w_i x_i < \theta_i + \varepsilon_2 \\ s_i &= -1 \text{ iff } \theta_i - \varepsilon_2 < w_i x_i < \theta_i - \varepsilon_1 \\ s_i &= \quad 0 \text{ otherwise.} \end{aligned} \tag{10}$$

The choice of ε_1 and ε_2 is critical to the success of the algorithm. In the absence of theoretical arguments to guide the choice of thresholds, it is encouraging that the same ε_1 and ε_2 were found to be adequate on a wide range of data sets including both spoken and handprinted numerals [Ide '66].

The initial hyperplanes in these experiments were generated by a supervised version of the same algorithm. Several interesting comparisons were obtained between continued training and unsupervised learning.

IV. Cluster Analysis

So far we have discussed assigning patterns to predetermined classes. In some problems, however, even the number and nature of the classes, if any, are unknown. How many distinct varieties of handprinted 2's are there? How many different types of clouds can one observe in satellite photographs? Do the accumulated electrocardiac records of thousands of patients contain a clue to the possible varieties of heart disease? Can the electrical activity monitored by a single electrode implanted in the optic nerve be analyzed to reveal the number of characteristics of active fibers? Would a two-level procedure effectively reduce search time in the automatic identification of Chinese ideographs? These and similar questions form the object of the range of techniques known as cluster analysis, numerical taxonomy, mode seeking, or unsupervised learning.

Aside from their common quest for some manner of grouping, the outstanding feature shared by the above questions is vagueness. The proper measure of similarity to be used for grouping samples cannot be rigorously deduced from the teleological guidelines offered in these applications. In the absence of an objective performance criterion, similar to the error rate in pattern classification, a universal standard cannot be formulated for clustering methods.

The examples in Fig. 7 illustrate both "easy" and "difficult" groupings in two dimensions; our modest goal is to find automatic methods for delineating the "easy" clusters in higher-dimensional spaces. The difficulty is that clustering is so much a "gestalt" operation, particularly well suited to the human being's ability to consider multiple relation-

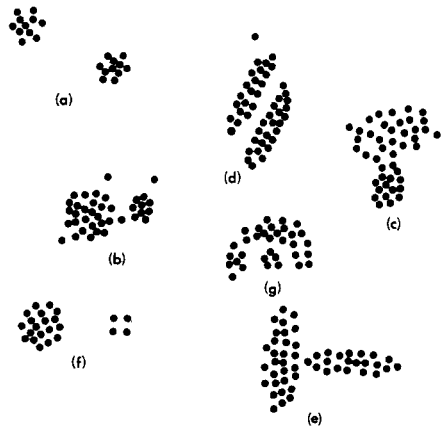


Fig. 7. Examples of clusters in two dimensions. Some of the difficulties encountered in delineating clusters are: bridges between clusters caused by strays (b) and perhaps (c); nonspherical covariance matrices (d); nonproportional covariance matrices (e); unequal cluster populations and spurious peaks in the projected distributions caused by small sample size (f); and linearly nonseparable clusters (g). In real applications all of these conditions may occur simultaneously; the ideal situation shown in (a) is the exception rather than the rule.

ships simultaneously. What a pity our power fails us in hyperspace!

In digital computer simulation, on the other hand, we must compute the pertinent relations pair by pair and component by component, and then study *their* relations to one another. Raising the dimensionality usually imposes no penalty other than a linear increase in processing time.

Distance Measures

Gestalt or not, the first step in clustering a data set is to define some measure, albeit arbitrary, of the *similarity* between two samples. In order to apply some of the previous concepts about the manipulation of objects in hyperspace to this problem, it is convenient to choose a measure of similarity with the properties of a *distance function*. These properties correspond to our intuitive notions of similarity with respect to symmetry and the triangle inequality. In any case, it is always possible to imbed the objects in a metric space in such a way that the rank order of the pairwise distances corresponds to that of the similarities [Shepard '62, Kruskal '64].

For patterns with numerical components, a simple distance measure is the mean square difference or Cartesian distance

$$d_{ij}^2 = (\bar{x}_i - \bar{x}_j)'(\bar{x}_i - \bar{x}_j).$$

This measure is proportional to the correlation between the two vectors normalized by the mean square length. When the relative sizes of the components have no intrinsic significance, it may be useful to transform the pattern space to normalize the mean square distance between the pattern pairs along every coordinate axis before computing the similarity relations [Sebestyen '62]. The relations of several other invariant criteria for grouping data have been studied both analytically and experimentally [Friedman '66, Rubin '66].

For binary patterns, the most common measure is the correlation $\bar{x}_i'\bar{x}_j$; this is sometimes normalized by the number of ones in one or the other of the vectors, or the arithmetic or geometric mean of the number of ones in both vectors. A somewhat more complicated measure with some desirable information theoretical properties but without all the attributes of a metric is the Tanimoto criterion [Rogers '60] advocated for clustering in [Ornstein '65]:

$$d_{ij} = -\log_2 \frac{\bar{x}_i'\bar{x}_j}{\bar{x}_i'\bar{x}_i + \bar{x}_j'\bar{x}_j - \bar{x}_i'\bar{x}_j}. \tag{11}$$

A number of other similarity measures are cataloged in an excellent survey on multidimensional data analysis [Ball '65].

Clustering Methods

The principal objective of cluster analysis is to gain more information about the structure of a data set than is possible by more conventional methods such as factor analysis or principal components analysis. The level of detail we can reach depends on the dimensionality of the data and the number of samples we propose to examine.

When it is necessary to minimize the number of distances computed, and there is reason to believe that the individual clusters are tight and widely spaced, the *chain* method may succeed [Bonner '62, '64]. Here the first sample is taken as a representative of the first cluster. The distance of the second sample to the first sample is computed, and if this distance exceeds a preset threshold, a second cluster is started. Otherwise, the second sample is also included in the first cluster. In like fashion the distance of each new sample to a representative of every established cluster is thresholded, and a new cluster is started only if none of distances is below threshold. It is not, of course, difficult to think of examples where this procedure fails.

When sufficient computing power is available to calculate the distance between all of the $\frac{1}{2}N^2$ pattern pairs, the *similarity matrix* approach gives better results. Here also the distances are thresholded. The similarity matrix is a symmetric $N \times N$ matrix which contains ones in the entries corresponding to below-threshold pattern pairs, and zeros elsewhere. The similarity matrix is now considered as the characteristic matrix of an *undirected graph*, of which all the disjoint subgraphs must be determined [Abraham '62]. There are several algorithms which do this efficiently [Baker '62, Warshall '62].

With 1000 patterns of 100 components each, which constitutes a relatively small problem, 5×10^7 multiplications and additions are required to compute the similarity matrix.

Thresholding the distances is a rather coarse operation, and in general better results can be achieved by working with the numerical values. When the expected number of clusters is known, a suitable objective is to partition the data set in such a way that the average scatter of the clusters is a minimum. Fortunately, there is a family of simple iterative algorithms which guarantee at least a local minimum in the mean distance of each sample from the closest cluster center [Ball '66, MacQueen '67, Casey '67[1], Dorofeyuk '66, Braverman '66[2]].

Let us denote the distance between two vectors $\bar{a}$ and $\bar{b}$ by $d(\bar{a}, \bar{b})$, let the set of pattern vectors comprising the kth cluster at the jth step be C_j^k, and let the corresponding "cluster center" be $\bar{c}_j^k$. The two steps of the algorithm are:

1) Assign every pattern vector x_i to one and only one cluster

$$\bar{x}_i \in C_j^k \text{ iff } d(\bar{x}_i, \bar{c}_j^k) = \min_{\ell} d(\bar{x}_i, \bar{c}_j^{\ell}). \tag{12}$$

2) Define new cluster centers

$$c_{j+1}^k \ni \sum_{\bar{x}_i \in C_j^k} d(\bar{x}_i, \bar{c}_{j+1}^k) = \min_{\bar{y}} \sum_{\bar{x}_i \in C_j^k} d(\bar{x}_i, \bar{y}).$$

The initial cluster centers may be specified by one of the procedures mentioned earlier, or even set at random. For $d(\bar{a}, \bar{b}) = |\bar{a} - \bar{b}|$, the second step reduces to computing the mean vector of each cluster. Then the distance from every pattern to each new cluster center is computed and the cycle is completed by reassigning the patterns where necessary. Since the mean distance between the patterns and the appropriate cluster center decreases in *both* steps, convergence is certain.

Fig. 8 shows the action of the algorithm on three clusters. With arbitrary initial cluster centers, convergence takes three cycles; with a more fortuitous start, one cycle would be sufficient.

The choice between a *hierarchical* and a *single-level* procedure depends on additional knowledge about the structure of the sample set. Thus, we could partition a set into 32 clusters either directly or by successive dichotomies. Hierarchical methods are particularly appropriate in taxonomic applications [Michener '57, Sokal '63]. It is also sometimes advantageous, in order to conserve processing time, to split the data into a number of very small groups, and then treat each group as a single sample as represented by its mean vector.

In some applications *overlapping clusters* are permissible; a single pattern may be assigned to several clusters. Algorithm (12) is still applicable if in step 1) the pattern assignment is determined by thresholding rather than minimization.

Mavericks, or patterns mutilated by the transducer, bursts of noise, keypunch errors, and the like, present a far more serious problem here than in conventional recognition methods. When they are far removed from the main body of data, they monopolize cluster centers, while if they fall between legitimate clusters, forming bridges, they may cause spurious mergers. Special rules, aimed at eliminating pocket boroughs at one end of the scale, and cartels at the other, are required to deal with these anomalies.

Heuristics are also useful when the number of clusters is not known in advance. Here new clusters must be created to accommodate samples far from the existing ones, and old clusters must be destroyed when their members have been taken over by the new constellations. The "birth" and "death" processes are usually based on a "one change at a time" philosophy which can lead to trapped states, but anything more complicated is computationally prohibitive [Ball '66, MacQueen '67].

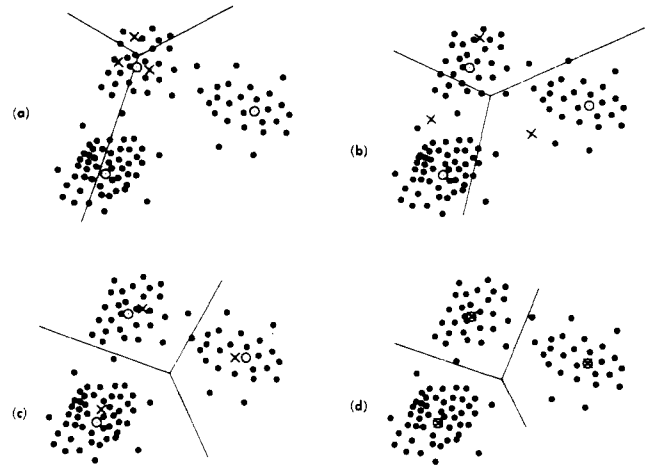


Fig. 8. A convergent clustering algorithm. The black dots are the pattern vectors, × is the variable cluster center, and the solid lines represent the partitioning boundaries. At each step the cluster center moves to the center of gravity of the corresponding partitioned samples. This defines a new linear boundary halfway between each pair of cluster centers. The process terminates when the cluster centers become coincident with the true centers of gravity of the clusters (indicated by circled dots).

In general, algorithms based on the minimization of a distance function [Ball '66, Fortier '65, MacQueen '66] are most appropriate for fairly isotropic clusters [Fig. 7(b), (c), (e), and (f)], while methods maximizing the minimum distance between the members of two distinct clusters [Bonner '62, Abraham '62] are good for dense, clearly separated clusters of whatever shape [Fig. 7(d) and (g)].

V. Feature Extraction

In some pattern classification problems (and also in data sets intended for cluster analysis) the patterns in the various classes (clusters) are so intermixed that only a highly nonlinear method can separate them to the required degree of accuracy. Rather than resort to the few nonlinear algorithms available, some workers prefer to arbitrarily divide the process into two stages, the first of which consists of simplifying the problem sufficiently to render it tractable for the second. This is particularly helpful in multiclass problems.

The design of the first, or feature extraction, stage may be approached from two points of view. Either one attempts to transform the sample space in such a manner that the members of each class exhibit less variability and the relative separation between the classes is increased, thus allowing the use of a simpler decision mechanism, or one reduces the dimensionality of the sample space, permitting the application of more complicated decision schemes. Ideally one could accomplish both objectives with the same transformation, but unfortunately the transformations appropriate for the first objective generally increase the dimensionality.

The chief theoretical difficulty in feature extraction is that the features must be evaluated in terms of the decision stage rather than on their own. Convenient numerical criteria like error rate can only be used to evaluate the whole

system, thus the performance of the optimum (or best available) second-stage classifier should really be computed for every measurement procedure under consideration. However, some shortcuts are available, and will be mentioned in a later section.

Several current approaches to feature extraction in the broad sense will now be reviewed without much regard for a common theoretical framework.

Preprocessing

The first step in preprocessing, *object isolation*, is all too often ignored in laboratory studies. Yet touching characters are responsible for the majority of errors in the automatic reading of both machine-printed and handprinted text, the segmentation of speech remains an unsolved problem, and the recognition of chromosomes and blood cells, and of objects of interest is reconnaissance photographs, is greatly hampered by shortcomings in this area.

In simpler tasks, such as reading typescript, heuristic algorithms based on the relative uniformity of size, texture, or spacing of the patterns are often successful [Brain '66[1], Hennis '66]. In general, however, it seems that the isolation problem can only be solved by including it in a loop with the recognition process by trying different partitionings of the overall pattern until the individual components "make sense." The heuristic techniques involved here are now becoming known under the name of *scene analysis*.

More success has been achieved in *noise filtering*. Simple filters remove isolated dots in scanned patterns or extraneous noise outside the normal frequency range of speech, while more sophisticated filters can restore smudged areas in fingerprints and fill in long gaps in particle tracks in bubble chamber photographs [Wadsworth '66]. Averaging operations are used to good advantage in studies of evoked-response electroencephalograms and single-fiber preparations [Moore '65]. The noise encountered in pattern recognition applications (as elsewhere!) is seldom white; unwelcome correlations are introduced through improper registration and limitations of the transducers.

Size and shear normalization for two-dimensional patterns can be carried out rather elegantly by applying a linear transformation which transforms the 2×2 moment matrix of the pattern into the identity matrix [Casey '67[2]].

The corresponding *time scale* normalization in speech is more difficult, since a linear transformation is usually insufficient. Variations in the relative lengths of the different segments of a word depend not only on the speaker, but also on the precise nuance he wishes to express [Beetle '67].

Amplitude normalization may be accomplished by automatic gain control (AGC) of the overall speech waveform. More sophisticated methods use ratios of the frequency components, usually obtained by taking logarithmic differences.

Complex images can be transformed by the iterative application of simple *neighborhood operators* whose output at each step depends only on the values of their nearest neighbors. Propagation effects ensure that the final value at each point comprises information originating at every point of the original picture. These methods are suitable for following tracks (as in cloud chamber photographs), tracing ridges and valleys, defining boundaries, and obtaining certain other geometric and topological features of the patterns [Narasimhan '64]. For binary patterns, a *skeleton* characterizing the general shape of the pattern may be conceived of as the locus of mutual extinction of "grass fires" simultaneously ignited at every zero-one transition in the image [Blum '67, Rosenfeld '66, Pfaltz '67].

Contour information may also be obtained more directly by means of gradient methods. A convenient technique of scanning *line drawings* (such as handprinted characters) is provided by the flying-arc curve follower. Here the output is a coded version of the path followed to track the curve, rather than a point-by-point representation of the whole retina [Greanias '63].

Registration

There are at least three common approaches for taking advantage of the fact that in some pattern recognition problems, such as character and spoken word recognition, some members of each class differ from one another only by a translation of the coordinate axes.

The first approach is *preregistration*. In character recognition *edge* registration (after removal of stray bits), *center of gravity* registration, and *median* registration (which is faster on a digital computer) have been used. In speaker verification it is customary to select phrases with an initially rapidly rising envelope to trigger the sampling apparatus.

Another approach is to try all the measurements, whatever they are, in every possible position, and use the number of "hits" or a thresholded function thereof, in the final feature vector. This is the path taken in Rosenblatt's *similarity constrained perceptron* [Rosenblatt '62[1]], and in many template matching schemes. There is some physiological evidence to show that the mammalian visual cortex also contains cells which are responsive to the same feature occurring almost anywhere on the retina [Hubel '62]. An efficient hardware implementation for binary patterns consists of shifting the digitized patterns through a one- or two-dimensional shift register, and using the outputs of the cells to drive the measurement logics [Andrews '62, Booth '62, Griffin '62, Rabinow '62]. Fig. 9 shows this configuration.

The third approach consists of using *translation invariant* measurements. The best-known class of translation invariant measurements are the *autocorrelation functions*. The first-order autocorrelation function, defined in one dimension by

$$\phi_x^1(\tau) = \int x(t)x(t+\tau)dt, \tag{13}$$

was first applied to character recognition in 1958 [Horwitz '61]. Its main shortcoming is that it is not a one-to-one transformation; several patterns, differing in more than just translation, may have the same autocorrelation function.

This difficulty is avoided by use of *higher-order auto-*

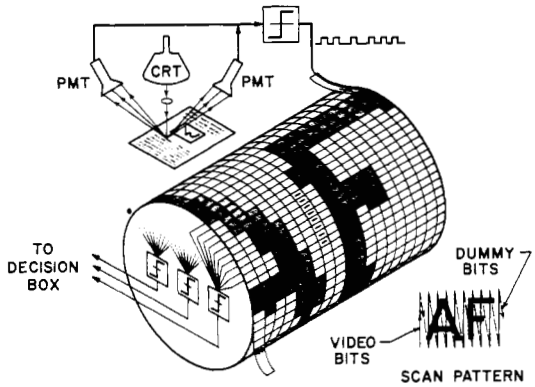


Fig. 9. Cylindrical representation of a shift register for two-dimensional patterns. The shift register, conveniently thought of as wrapped around a cylinder, has direct (+) and inverted (−) outputs to allow the extraction of features comprising both black and white points. The feature shown is responsive to white-black vertical edges over the whole field. Up to 100 such measurements may be coupled to the register. The buffered outputs serve as the input to a linear categorizer or other decision device.

correlation functions, which can be shown to be unique, for $n \geq 2$, except for translation [McLaughlin '67]:

$$\phi_x^n(\bar{\tau}_1, \bar{\tau}_2, \cdots, \bar{\tau}_n) = \int x(\bar{t})x(\bar{t} + \bar{\tau}_1)x(\bar{t} + \bar{\tau}_2)\cdots x(\bar{t} + \bar{\tau}_n)d\bar{t}. \quad (14)$$

If the inner product is used for classification in the autocorrelation space, the immense amount of computation necessary to obtain these transforms for large n can be reduced through the relation

$$\int \phi_x^n \phi_y^n d\bar{\tau}, d\bar{\tau}_2, \cdots, d\bar{\tau}_n = \int \left[\int x(\bar{t})y(\bar{t} + \bar{\tau})d\bar{t} \right]^{n+1} d\bar{\tau}. \quad (15)$$

Even this calculation is impracticable for a large number of patterns and classes, but good recognition results have been recently obtained on small alphabets [McLaughlin '67]. It seems even the simple correlation technique is adequate for classification in these very high-dimensional spaces.

Other translation invariant functions which have been used to characterize patterns are the *Fourier transform* (here also computational shortcuts are available [Cooley '65]) and higher-order *central moments* [Alt '62, Hu '62]. Unfortunately they lack the desirable property of uniqueness.

Invariants can also be derived for scale changes and for rotation.

Intuitive Measurements

An enormous amount of ingenuity has been applied over the years to devising "good" measurements for various pattern recognition tasks. Since the measurements favored by different investigators are seldom compared on the same data sets, objective evaluation of the merits of the different systems is difficult. The following is merely a sampling of the range of techniques available.

Simple *geometric features*, such as straight lines, edges, arcs, corners, and circles of various sizes, can be detected with mask-type threshold devices (Fig. 10). *Topological* information, such as the number of line segments encountered by a slice of specified orientation through the pattern and the existence of bays, enclosures, indentations, and symmetry conditions, is easily obtainable by means of logical tests on a digital computer [Doyle '60, Freeman '62, Kuhl '63, Perotto '63, Clemens '65, Munson '67].

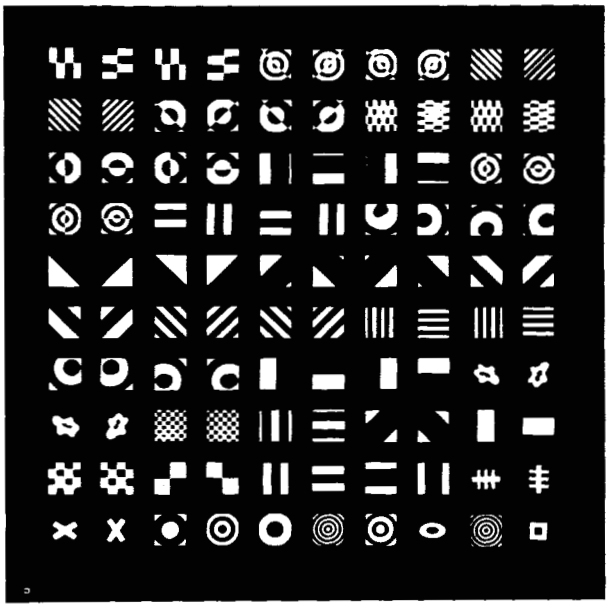


Fig. 10. Masks for extracting geometric features. One hundred features are obtained simultaneously by means of this mask plate, a fly's eye lens assembly, and a set of photocells. A more recent version of the apparatus with 1000 apertures is responsive to elementary features such as black-white edges in various orientations in different regions of the object image. The device was developed at the Stanford Research Institute under the sponsorship of the U. S. Army Electronics Command.

Stroke information is valuable in character recognition, especially on handprinted letters and numerals [Bernstein '64, Groner '67]. The location and kind of horizontal and vertical *extrema* have also been used to good advantage [Teager '65].

In tracking particles in cloud chamber and bubble chamber photographs, the intersections of tracks, and their degree of curvature, are significant. Intersections and end points can be readily found by means of iterative *neighborhood operators* of the kind already mentioned [McCormick '63].

Formant extraction, or localization of the ridges in the energy-frequency-time spectrum, is a common technique in speech recognition. To eliminate variations due to slow or fast speech, samples can be obtained only following marked transitions [Flanagan '65].

Texture information, or the relative constancy of gray levels, has been used to separate woodlands, cultivated areas, urban centers, and lakes in aerial photographs [Hawkins '66]. The size and location of the peaks in the *gray level histogram* of blood cells is sufficient to discriminate several types [Prewitt '67].

Heuristic methods such as these are largely responsible for almost all of the pattern recognition devices which have been incorporated in practical systems to date.

Random Measurements

We have seen that a nonlinear decision boundary in the original sample space appears as a linear boundary in the space spanned by the products of the original variables. In general, nonlinear combinations other than products also exhibit the same effect. It is often possible to solve linearly nonseparable problems, without increasing the dimensionality, by means of a random nonlinear transformation.

One example of such a transformation is furnished by the layer of *A-units* in the simple perceptron (Fig. 11). Each A-unit implements a threshold function. It has been shown that the relative effectiveness of the A-units is a strong function of the parameters guiding the random assignment of input points [Rosenblatt '62]. Some problems, particularly those involving topological distinctions, cannot be solved with any assignment [Papert '67].

Random transformations are also helpful in reducing the statistical dependencies among the original variables. Random *n-tuples* have been used by Bledsoe and Browning, and by Kamentsky and Liu, in improving the performance of maximum likelihood classifiers [Bledsoe '59, Kamentsky '63]. In these instances the improvement was obtained despite a significant decrease in the dimensionality of the space.

When we see the gains obtained with random features, we inevitably wonder how much more we could improve matters by judicious selection. This brings us to the subject which follows.

Selection Algorithms

We must first dispose of the expectation that there exists a selection algorithm by means of which we can find an optimal set of measurements of the type discussed above. Let us consider 7-tuples defined on 20×20 binary patterns. Then, even if we restrict ourselves to sets of only 100 measurements we must evaluate no less than

$$\binom{400}{7}_{100} \simeq 10^{1300}$$

distinct sets!

Of course, even with the fastest computers, we cannot hope to sample more than an infinitesimally small fraction of all possible *n*-tuples. This argument is sometimes used against expanding the search to more complex, and therefore more numerous, measurements, such as threshold functions.

The measurements should not be selected individually without regard for other measurements already included or about to be included in the set, because we might be extracting redundant information, or even duplicating measurements. Nevertheless, most selection schemes assume that the measurements are independent. Because of the large number of measurement sets to be tried in order to even skim those available, an easy-to-evaluate criterion is essential.

An *information measure* used by Liu to select *n*-tuples for consideration is

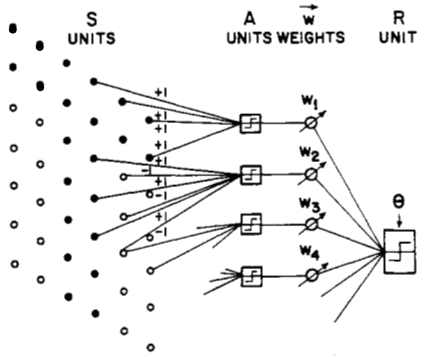


Fig. 11. Simple perceptron. The input vector appears on the layer of sensory units. Feature extraction may be considered as taking place in the associative layer. The result of a linear categorizer scheme is the output of the response unit. Although the simple perceptron was originally intended merely as the next step above a single threshold unit in a hierarchy of functional blocks to be used in brain modeling, it has been extensively applied to pattern classification tasks. The optimal assignment of origin points and thresholds for the A-units remains an unsolved problem.

$$I(y_k) = \lg m + P(y_k = 0) \sum_{i=1}^{m} P(C^i|y_k = 0) \lg P(C^i|y_k = 0) + P(y_k = 1) \sum_{i=1}^{m} P(C^i|y_k = 1) \lg P(C^i|y_k = 1) \quad (16)$$

where y_k is a binary measurement and m is the number of equiprobable classes.

This gives a measure of the number of class pairs separated or partially separated by the *n*-tuple y_k. Its value is 1 if the measurement is always "on" for half the classes and "off" for the other half, and 0 if it is "on" with the *same* probability for every class.

In making up the final measurement set, the information measure I is supplemented by the *pairwise distance* (not really a distance at all!)

$$D_k^{ij} = |\lg P(y_k|C^i) - \lg P(y_k|C^j)| \quad (17)$$

which is summed over k to yield an indication of the class pairs most in need of additional separating power [Liu '64].

Another measure of pairwise separation is the *mean-to-variance ratio* [Bakis '68].

$$F_k^{ij} = \frac{(P_k^i - P_k^j)^2}{P_k^i(1 - P_k^i) + P_k^j(1 - P_k^j)} \quad (18)$$

where

$$P_k^i = P(y_k = 1|C^i).$$

All three of these measures can be used either to build up a measurement set from a pool of available candidates, or to pare down the pool by dropping the least promising members. Estes has shown that under certain liberal assumptions the two procedures lead to equivalent results [Estes '64]. There is ample empirical evidence that such methods are far superior to purely random selection, although they are not as good as selection by means of a direct performance criterion.

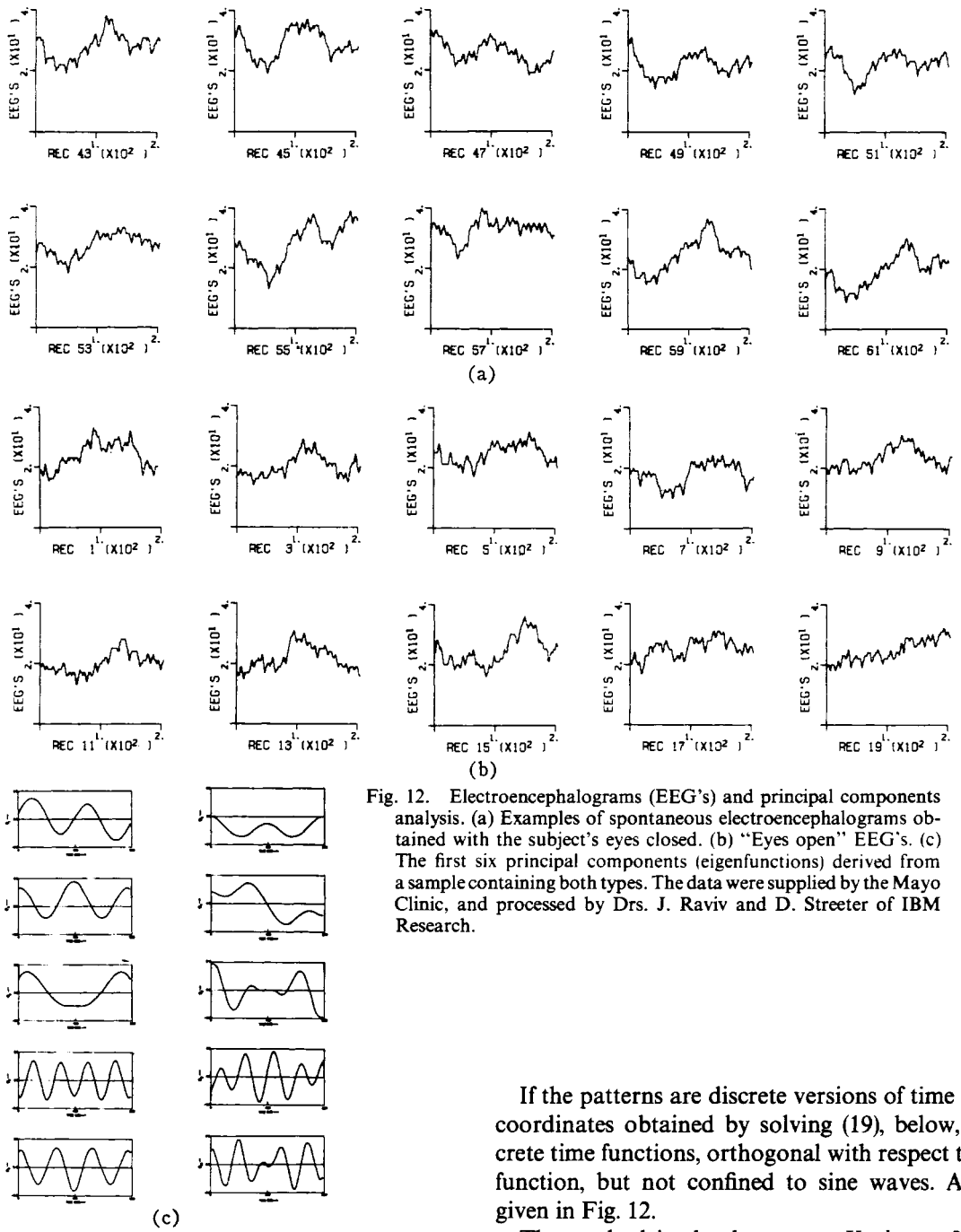


Fig. 12. Electroencephalograms (EEG's) and principal components analysis. (a) Examples of spontaneous electroencephalograms obtained with the subject's eyes closed. (b) "Eyes open" EEG's. (c) The first six principal components (eigenfunctions) derived from a sample containing both types. The data were supplied by the Mayo Clinic, and processed by Drs. J. Raviv and D. Streeter of IBM Research.

Dimensionality Reduction

Although some of the feature extraction methods we have examined result in a decision space of dimensionality lower than the original sample space, the phrase "dimensionality reduction" is usually reserved for linear transformations.

The easiest approach, that of *principal components* analysis, is to ignore the labels of the samples [Wilks '60]. The transformation then emphasizes the directions in which the samples show the greatest scatter. This is equivalent to finding the subspace in which the original vectors may be approximated in the *least mean square error* sense. It may also be thought of as *generalized spectral analysis.*

If the patterns are discrete versions of time functions, the coordinates obtained by solving (19), below, are also discrete time functions, orthogonal with respect to a weighting function, but not confined to sine waves. An example is given in Fig. 12.

The method is also known as *Karhunen-Loeve* analysis and *principal factors*, and is sometimes included under *discriminant analysis.* The eigenvalue equation which must be solved is

$$(A - \lambda I)\bar{w} = 0 \tag{19}$$

where A is the sample covariance matrix.

The basis of the transformed space consists of the eigenvectors associated with the largest eigenvalues of (19). If we wish to transform down to m dimensions, the eigenvectors of the m largest eigenvalues are taken. To find the coordinates of a given sample, it is simply projected onto the eigenvectors:

$$y_{ik} = \frac{\bar{x}_i' \cdot \bar{w}_k}{|\bar{x}_i|}$$

where y_{ik} is the kth coordinate of the ith sample in the transformed space and $\bar{w}_k$ is the eigenvector associated with the kth eigenvalue of (19).

A disadvantage of this procedure is that it does nothing to preserve the separation between the classes. As we eliminate the information contained in the subspace orthogonal to the reduced space, we may be throwing the baby out with the bathwater.

To take into account the distribution of the classes, we may resort to the generalized eigenvalue method presented in Section II, the subsection on "Discriminant Analysis," and use several eigenvectors, instead of just the first [Sebestyen '62, Raviv '65].

A more sophisticated procedure, which attempts to minimize the error rate obtained in the reduced space with linear discriminants rather than just the ratio of intraclass scatter to interclass scatter, has been developed [Casey '65[1]]. The transformation is obtained by means of a gradient technique on a power series expansion of a lower bound on this error rate. It is assumed that the projection of the samples on the normal to the optimum hyperplane has a Gaussian distribution. This assumption is somewhat less restrictive than requiring Gaussian distributions to begin with.

Linear dimensionality reduction techniques are most advantageous when the number of pattern classes is large. Otherwise, computing the linear combinations for each sample offsets any gain in computation which may be obtained in the decision stage.

VI. Experiments in Pattern Recognition

Almost all of the methods discussed in the previous sections have been subjected to a certain amount of experimental investigation, mainly through digital computer simulation. Nothwithstanding, really large-scale, well designed experiments on realistic data sets are still the exception rather than the rule.

This section represents an attempt to give the reader some idea of the nature of the experimental data sets and of the results obtained in several possible applications of pattern recognition techniques. Although only one example has been selected from each area, the degree of achievement in the various applications may be taken as an indication of inherent difficulty rather than of the relative skill of the experimenters. One of the main accomplishments of the past decade has been, in fact, the establishment of a gross ranking of the feasibility of proposed recognition tasks.

We have seen that, in a large number of applications, obtaining suitable data in computer-readable form is the main source of difficulty. In others, as in electroencephalography and other biomedical applications, evaluation of the results is the critical problem; comparison to human standards, especially at intermediate stages of processing, is by no means necessarily the most suitable criterion.

It is hoped that the rather sketchy descriptions of actual experiments which are to follow will serve to emphasize these and other procedural difficulties common to large segments of pattern recognition, as well as to illustrate positive achievements.

Impact Printed Characters

The study described here [Liu '66] reports performance figures on a data set comprising about 70 000 typewritten characters (Fig. 13) in 13 different type styles, with a full range of ribbon life, and several document sources for each font.

Double-spaced samples of both upper and lower case are scanned on a cathode ray tube scanner controlled by a small general-purpose computer. The scanning program takes care of character localization, character centering, and linewidth control by means of video threshold adjustment. Ninety-six n-tuple measurements are extracted with special-purpose digital hardware. The resultant feature vectors are categorized by means of a variety of decision algorithms.

Error rates range from 0 percent substitution errors with 0.1 percent rejects on fonts which were represented in the design material to 7.5 percent errors and 18 percent rejects on new fonts with marked stylistic eccentricities.

A two-level decision scheme is used to conserve computation time. It is shown that the number of candidates retained in the first level affects the overall error by several orders of magnitude. Multiple hyperplanes per class specified with low accuracy (few bits) are compared to single hyperplanes specified with high accuracy. The piecewise linear boundary is superior when the total number of bits of storage is the same; this is no doubt due to the multimodal structure of the data.

The authors properly point out that although a large data set was used, the results cannot be readily interpolated to expected recognition rates in the field, where segmentation errors, poorly adjusted typewriters, overstrikes, and document mutilation would result in markedly inferior performance. The usefulness of the study must be seen in terms of its application to the optimization of parameters in a practical multifont reading machine.

Handprinted Characters

Repeated demonstrations have shown that without context even human recognition of handprinted alphanumerics is surprisingly poor. It takes considerable courage just to tackle the problem in its full generality, without restricting the material to one or a few writers or imposing severely handicapping stylistic constraints.

One inexhaustible source of data consists of the output of computer programmers; in the investigation reported here [Brain '66[2]] FORTRAN coding sheets are used, as shown in Fig. 14. It is expected that copy produced under actual working conditions would be considerably worse than the test material, but even these ordered alphabets offer sufficient challenge for the initial phases of the investigation.

Some 8000 characters, representing 10 repetitions of the 46-character FORTRAN alphabet by 16 writers, are scanned with a television camera and quantized on a 24×24 bit

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Hermes-Techno Elite

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Selectric Elite

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Royal Manual Standard Elite

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Selectric Adjutant

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

IBM Model B Artisan

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

IBM Model B Dual Basic

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

IBM Model B Courier

V W X Y Z A B C D E F G H I J K L M N O P Q R S T U

v w x y z a b c d e f g h i j k l m n o p q r s t u

Olympia Senatorial

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Selectric Delegate

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

IBM Model B Prestige Elite

F G H I J K L M N O P Q R S T U V W X Y Z A B C D E

f g h i j k l m n o p q r s t u v w x y z a b c d e

Remington Regal Small

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

IBM 403

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

IBM 403 Inverted

Fig. 13. Typefaces used in a multifont character recognition experiment. These fonts were selected because they span a considerable fraction of the variations encountered in a normal business environment. The sample includes serif and sans-serif characters, roman and gothic fonts, "special effect" styles, as well as exaggerated aspect ratios.

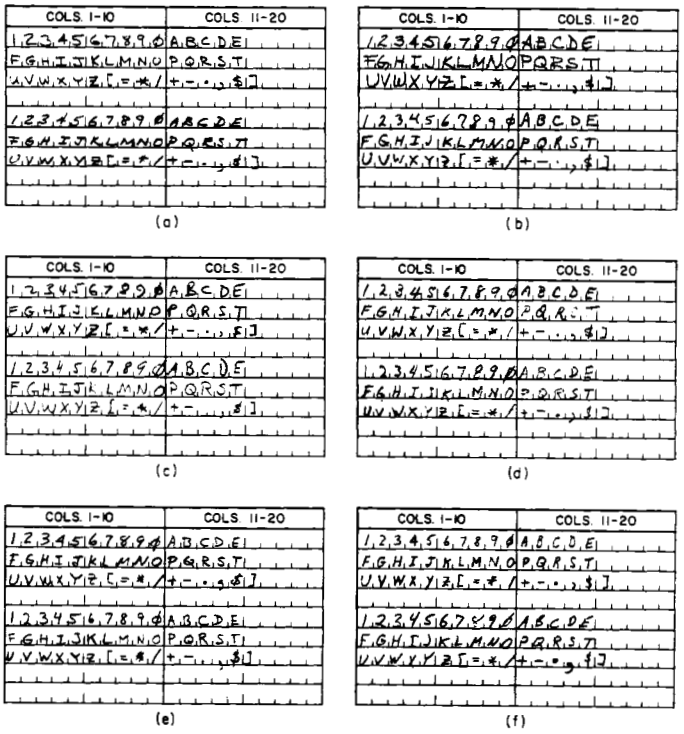


Fig. 14. Handprinted characters on FORTRAN coding sheet. An augmented FORTRAN alphabet is shown by each of twelve different writers. The range of variation is considerable even though the writers were in no particular hurry. These data were collected at the Stanford Research Institute under sponsorship of the U. S. Army Electronics Command.

field. Eighty-four simulated edge detectors are applied to the patterns in each of 9 translated positions. The sets of 9 feature vectors form the input to a trainable linear classifier.

The lowest error rates obtained on characters generated by writers not represented in the training set are of the order of 20 percent. Without translation, this error rate is roughly doubled.

Further experiments are planned to improve the feature detectors, to evolve "specialist" hyperplanes for difficult class pairs, and to generate new heuristic measurements. First results with these techniques are only moderately encouraging, with error rates of 15 percent for authors not in the training set, and 3 percent for "individualized" systems. It is believed that it is necessary to obtain a recognition rate of about 95 percent to allow a FORTRAN-oriented contextual error correction scheme to boost it to the ~99 percent required to compete with keypunching.

Automated Cell Image Analysis

CYDAC (CYtophotometric DAta Conversion) is a general-purpose sensor and transducer for microscopic images, developed at the University of Pennsylvania for quantitative studies of blood cells, chromosomes, and similar preparations [Prewitt '66]. The instrument consists of a flying spot scanning microscope, a photometer, an analog-to-digital data converter and associated digital recording equipment, and operator controls and monitoring systems.

The resolution of the scanner is about 0.25 micron in the plane of the specimen and the optical density is measured on a 256-level gray scale. A 50-by-50-micron field is sensed; thus for a single sample 40 000 eight-bit observations are recorded.

For the classification of leukocytes (white blood cells; see Fig. 15) it appears that the frequency histogram of the optical density of the entire cell contains enough information to identify the four major types sought in the study reported by Prewitt. The components of the histogram due to the nuclear and cytoplasmic regions are readily isolated by finding the local minima in the distribution. Shape information would be redundant.

Altogether 35 different measurements were investigated in the study. Examples of these are: cell mean optical density; skewness of the nuclear optical density distribution; nuclear optical density range; kurtosis of the cytoplasmic optical density distribution; standard deviation of nuclear optical density; and other derived parameters of the overall optical density distribution. No single one of these parameters is sufficient to separate all four cell types, but four pairs and 21 triplets are adequate for linear discrimination of all the samples.

The 50 leukocytes used in the study are sufficient for results significant at the 95 percent confidence level.

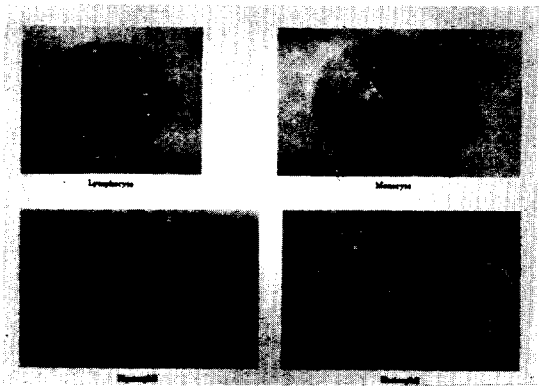


Fig. 15. Four types of white blood cells. These slightly distorted photomicrographs show the differences in the areas of the image at various levels of the gray scale which allow the cell types to be reliably differentiated with relatively simple measurements. Permission to use this illustration was granted by Mrs. J. M. S. Prewitt of the University of Pennsylvania.

Particle Tracking in Bubble Chambers

PEPR (Precision Encoding and Pattern Recognition) was initiated at the Massachusetts Institute of Technology by Pless and Rosenson in 1961 to automate the laborious process of locating and measuring particle tracks in processing bubble chamber photographs. This tool is widely used for studying the production and interaction of high-energy nuclear particles; at present about 10 000 000 frames of film are processed annually in the United States. A single frame is shown in Fig. 16.

Although the PEPR system now requires manual prescanning of the photographs in order to locate the tracks, it is designed for eventual conversion to completely automatic operation. The most time-consuming portion of the operation is, in any case, accurate digitization. When the momentum of a particle must be determined from the curvature of a very short segment of track, measurements accurate to better than one part in 30 000 are necessary.

The precise position of the track is determined by collecting the light transmitted through the film from a defocused, astigmatic spot on the face of an accurately calibrated cathode ray tube. The orientation of the axis of elongation of the spot is swept through a full circle at every location tested; thus the system detects at once not only the presence of a line segment on the film, but also its direction.

Once the spot latches onto a track, a multilevel predictive program ensures that the track is followed with a minimum of wasted excursions. After rough identification of the track parameters, the step size is decreased, and the locations of the various portions of the track are measured to within about 25 microns. In addition to highly sophisticated optical and electronic design, this degree of precision requires computed polynomial interpolation with reference to a fiducial grid.

Details and additional references on this very large-scale endeavor in pattern recognition are available in [Pless '65] and [Wadsworth '66].

Programmed Terrain Classification

The study described explores automatic detection of orchards, wooded areas, lakes, oil tank farms, and railroad yards in aerial photographs taken at 1:50 000 scale [Hawkins '66].

Since a typical 9-by-9-inch aerial photograph contains upwards of 10^9 bits of information, it is advantageous to avoid storage of the digitized video inside the computer by extracting the relevant features with a programmable flying spot scanner. The scanner looks at overlapping $\frac{1}{4}$-inch squares brought within range of the cathode ray tube by an x-y table which moves at 70 steps per second. In each position of the table 256×256 addressable points are available. At each point the video is quantized into 16 gray levels.

Examples of the "masks" used for processing the images are shown in Fig. 17. 1024 mask points, divided into any number of masks, can be accommodated by the system. At each mask point the optical density is multiplied by the corresponding weight. The results are accumulated for each mask and thresholded. The masks are shifted over the whole subregion.

Heuristic algorithms based on the frequency and spacing of satisfied (above-threshold) masks are used to determine the contents of each subregion. The thresholds are established on the basis of the average gray level in each region, and may be readjusted during operation.

Although separate design and test sets were not used in this study, the authors claim about 85 percent correct

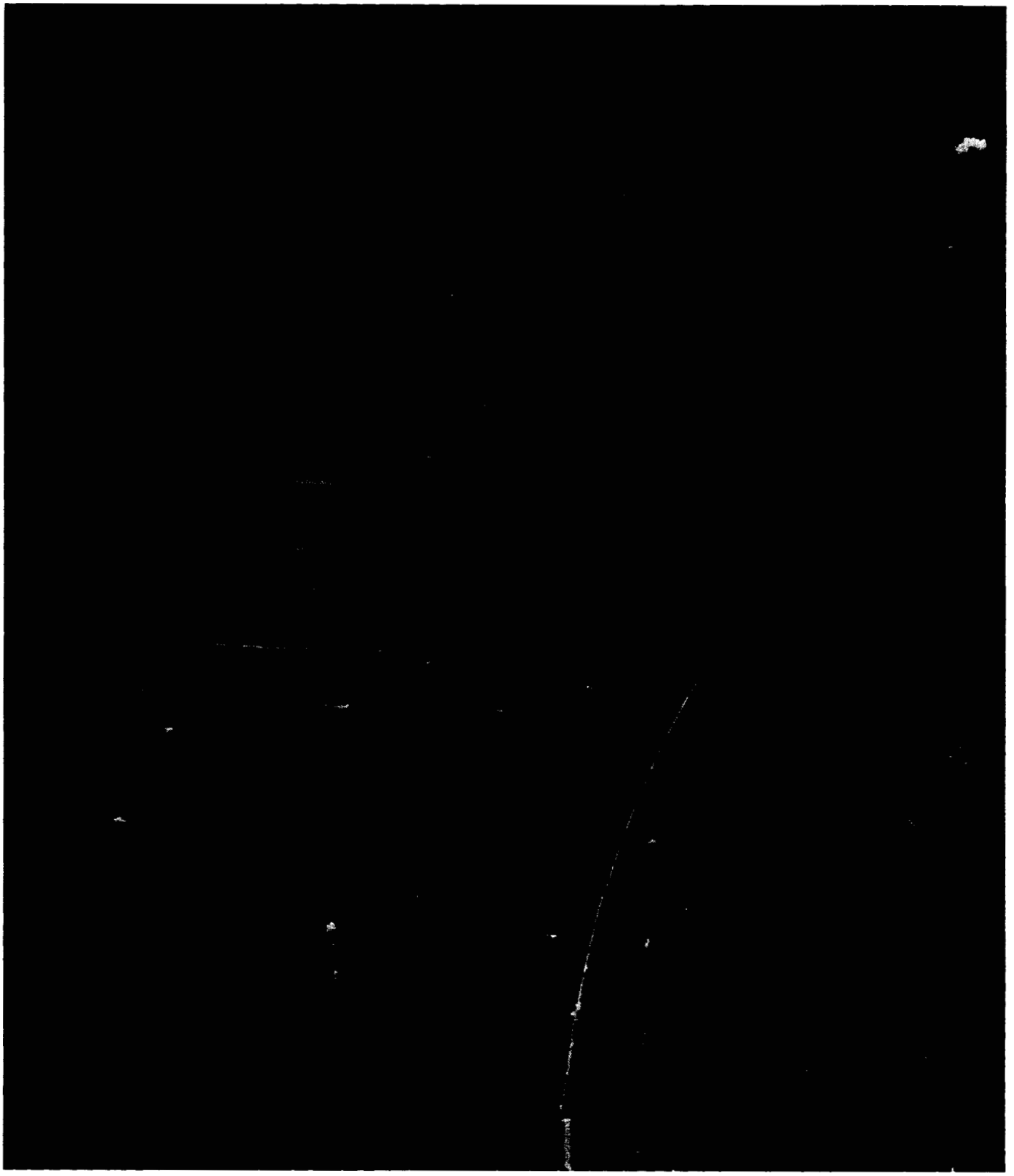


Fig. 16. A bubble chamber photograph showing particle tracks. The discontinuous nature of the tracks and the noisy background is inherent in the process. Two or three simultaneous exposures are generally obtained, for three-dimensional localization of the tracks. Note the fiducial marks. The photograph was obtained through the courtesy of the Stanford Linear Accelerator Center.

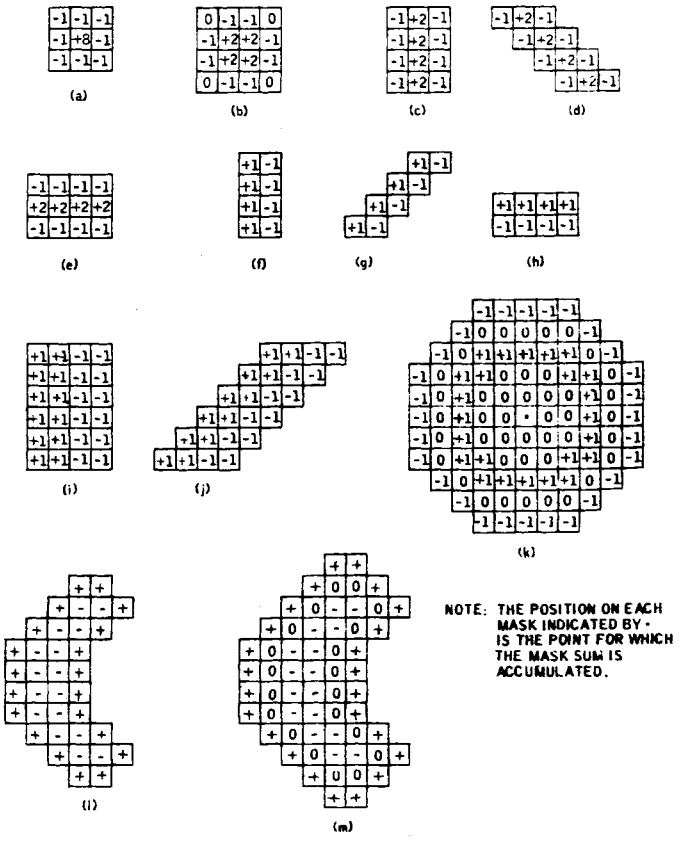


Fig. 17. Programmed masks for terrain classification from aerial photographs. These masks were designed to detect small dots, vertical, horizontal, and tilted edges and straight lines, and arcs. The masks are correlated with small subregions of the patterns. A mask is considered "on" whenever the correlation exceeds a certain threshold. These masks are used in experiments on photointerpretation at the Ford-Philco Corporation.

recognition over all the classes in 1124 classification decisions. The experimental system, while still much too slow for realistic evaluation, is an improvement over previously reported studies on only one or two photographs or on completely synthetic data.

Teleseismic Event Classification

In this study [Manning '66] nuclear explosions are differentiated from earthquakes on the basis of the characteristics of the compressional or *P*-wave, which travels a refractive ray path through the earth's mantle. At distances above a few thousand miles, which is the minimum range of the contemplated detection system, the *P*-wave arrives well ahead of the other components. Its dominant frequency is above that of the shear waves and surface waves, and seismic noise can be more easily eliminated from it by frequency filtering.

In general it is considered sufficient to differentiate explosions from shallow earthquakes, since deep earthquakes can be readily identified by the depth of focus. Seismograms of both shallow earthquakes and nuclear explosions are shown in Fig. 18.

The very small sample size available (38 events) renders the choice of measurements particularly difficult. Considera-

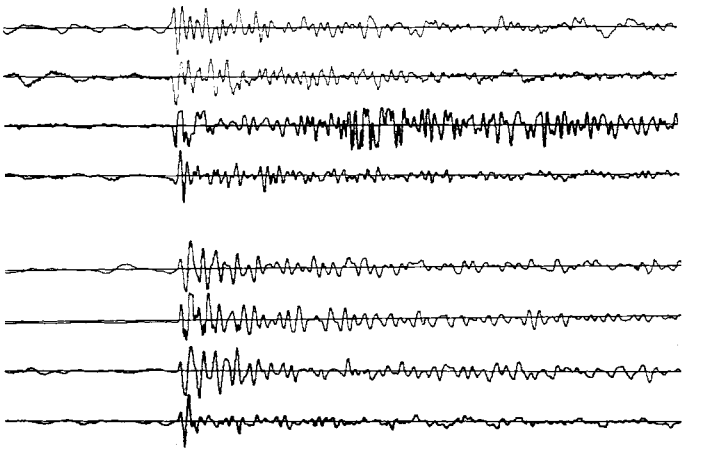


Fig. 18. Seismograms of earthquakes and nuclear explosions. The top four traces are seismograms of shallow earthquakes; the bottom four are underground nuclear explosions. The onset is clearly marked in every case. These seismograms were obtained from Bolt Beranek and Newman, Inc.

tion of the physics of the situation and a certain amount of computational experimentation led to the selection of average amplitudes in six 6-second time windows. These amplitude measures are shown to be superior for the task at hand to spectral measures involving zero-crossing rates.

The parameters of a maximum likelihood classifier were derived according to a normality assumption (see Section II) from ten samples of each of the two classes. Of the remaining events, 15 were classified correctly by this method. Equivalent results were obtained with human judgment aided by a display system which enables the operator to look at two-dimensional projections of the parameter space.

Electrocardiograms

Most attempts to automate the screening of electrocardiograms rely heavily on heuristic methods aimed at imitating the decision processes of trained electrocardiologists. Specht, on the other hand, reports considerable success with an elegant method based on nonlinear matched filters [Specht '67].

In one experiment 90 percent correct classification was obtained on "abnormal" traces, and 97 percent on "normal" traces. The training set in this experiment contained 200 patterns and the test set only 60 patterns, so the usual caveats about limited data sets apply.

By way of comparison, however, electrocardiologists basing their decision only on the ECG scored only 53 percent on the "abnormals" and 95 percent on the "normals." The "correct" decision was determined from several doctors' diagnoses based on the ECG, full clinical history, examinations, laboratory tests, and, sometimes, the autopsy.

The nonlinear matched filter corresponds to a curved decision boundary obtained by smoothing the sample distribution with a suitable filter "window." Such a boundary can accommodate even the multimodal distribution of "abnormals." Second-order terms turn out to provide a

sufficiently close approximation of the boundary polynomial; Specht's algorithm eliminates all but 30 of the 370 starting weights corresponding to combinations of the amplitude-time samples of the waveforms.

The method may be implemented either through a small digital computer or by means of a trainable hardware device. So far no comparisons with competing methods, on identical data sets, are available.

Spoken Words

The recognition of words in a continuous utterance is still only a remote possibility, but numerous experiments have been reported [Sebestyen '62, Sokai '63, Widrow '63, Flanagan '65, Tappert '66] on the classification of isolated spoken words and syllables. The ten digits form an often used vocabulary, although experiments on as many as 30 words have been attempted.

With a single speaker, performance on test utterances may reach 97 to 99 percent. When the investigator uses his own voice, special care must be exercised. In one experiment on adaptive speech recognition, for example, improvement of the system performance with time turned out to be caused by the speaker learning to modulate his voice in such a manner that his words were always correctly recognized by the unchanging (due to a "bug") decision parameters.

With a small group of speakers, whose previous utterances are used to train the classifier, recognition rates of 90 to 95 percent have been reported by several investigators.

When the speakers are chosen completely at random, and include females, foreign nationals, victims of the common cold, and other hard-to-deal-with species, performance falls to about 70 percent on the ten digits.

Some recognition schemes mentioned in Flanagan's book use the whole quantized energy-frequency-time spectrum of a word and obtain the decision by cross correlation with stored references. Others depend on detection of the principal formants. Time normalization is sometimes resorted to. Related experiments deal with speech compression for transmission and storage, speaker identification and verification, and the design of voice-actuated phonetic typewriters.

VII. Envoy

We have accompanied the reader into the labyrinth of pattern recognition, pointed out some of the salient landmarks, and offered battle on his behalf to the Minotaur of semantic difficulties. We hope that we have given him enough string to encourage him to return.

Acknowledgment

F. Rosenblatt is responsible for this author's interest in pattern recognition problems. Appreciation is also expressed for pleasurable and informative conversations on the subject with colleagues at the IBM Watson Research Center, especially R. G. Casey, C. K. Chow, J. Raviv, and G. Shelton; and with G. Ball, R. Duda, N. Nilsson, and C. Rosen of the Stanford Research Institute.

Bibliography

[1] [Abraham '62] "Evaluation of clusters on the basis of random graph theory," C. Abraham, IBM Research Memo., IBM Corp., Yorktown Heights, N. Y., November 1962.

[2] [Aizerman '64[1]] "The theoretical foundations of the method of potential functions in the problem of teaching automata to classify input situations," M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, *Automation and Remote Control*, vol. 25, pp. 821–838, June 1964.

[3] [Aizerman '64[2]] "The probabilistic problem of teaching automata to recognize classes and the method of potential functions," M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, *Automation and Remote Control*, vol. 25, pp. 1175–1191, September 1964.

[4] [Aizerman '65] "The Robbins-Monro process and the method of potential functions," M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, *Automation and Remote Control*, vol. 26, pp. 1882–1886, November 1965.

[5] [Allais '64] "The selection of measurements for prediction," D. C. Allais, Tech. Rept. 6103-9, Stanford Electronics Lab., Stanford, Calif., November 1964.

[6] [Alt '62] "Digital pattern recognition by moments," F. L. Alt, in *Optical Character Recognition*. Washington, D. C.: Spartan, 1962, pp. 153–180.

[7] [Anderson '62] "Classification into two multivariate normal distributions with different covariance matrices," T. W. Anderson and R. Bahadur, *Ann. Math. Stat.*, vol. 33, pp. 422–431, 1962.

[8] [Andrews '62] "Multifont print recognition," M. C. Andrews, in *Optical Character Recognition*. Washington, D. C.: Spartan, 1962, pp. 287–304.

[9] [Baker '62] "A note on multiplying Boolean matrices," J. J. Baker, *Commun. ACM*, vol. 5, February 1962.

[10] [Bakis '68] "An experimental study of machine recognition of hand-printed numerals," R. Bakis, N. Herbst, and G. Nagy, *IEEE Trans. Systems Science and Cybernetics*, vol. SSC-3, July 1968.

[11] [Ball '65] "Data analysis in the social sciences: What about the details?" G. H. Ball, *Proc. Fall Joint Computer Conf.* (Las Vegas, Nev.), pp. 533–559, December 1965.

[12] [Ball '66] "ISODATA, an iterative method of multivariate analysis and pattern classification," G. H. Ball and D. J. Hall, presented at the Internat'l Communications Conf., Philadelphia, Pa., June 1966.

[13] [Bashkirov '64] "Potential function algorithms for pattern recognition learning machines," O. A. Bashkirov, E. M. Braverman, and I. B. Muchnik, *Automation and Remote Control*, vol. 25, pp. 629–632, May 1964.

[14] [Beetle '67] "Flexible analog time-compression of short utterances," D. H. Beetle and W. D. Chapman, *Proc. 1967 Conf. on Speech Communication and Processing (Preprint)*, USAF Office of Aerospace Research, pp. 342–347.

[15] [Bernstein '64] "Computer recognition of on-line, hand written characters," M. I. Bernstein, Memo. RM-3758, RAND Corp., Santa Monica, Calif., October 1964.

[16] [Blaydon '66] "On a pattern classification result of Aizerman, Braverman, and Rozonoer," C. C. Blaydon, *IEEE Trans. Information Theory (Correspondence)*, vol. IT-12, pp. 82–83, January 1966.

[17] [Bledsoe '59] "Pattern recognition and reading by machines," W. W. Bledsoe and I. Browning, *Proc. Eastern Joint Computer Conf.* (Boston, Mass.), pp. 225–233, December 1959.

[18] [Block '64] "Determination and detection of features in patterns," H. D. Block, N. J. Nilsson, and R. O. Duda, in *Computer and Information Sciences*. Washington, D. C.: Spartan, 1964, pp. 75–110.

[19] [Blum '67] "A transformation for extracting new descriptors of shape," H. Blum, in *Models for the Perception of Speech and Visual Form*. Boston, Mass.: M.I.T. Press, 1967, pp. 362–380.

[20] [Bonner '62] "A 'logical pattern' recognition program," R. E. Bonner, *IBM J. Research and Develop.*, vol. 6, pp. 353–359, July 1962.

[21] [Bonner '64] "On some clustering techniques," R. E. Bonner, *IBM J. Research and Develop.*, vol. 8, pp. 22–32, January 1964.

[22] [Both '62] "Design considerations for stylized font characters," W. T. Both, G. M. Miller, and D. A. Schleich, in *Optical Character Recognition*. Washington, D. C.: Spartan, 1962, pp. 115–128.

[23] [Brain '66[1]] "Graphical-data-processing research study and experimental investigation," A. E. Brain and J. H. Munson, Final Rept., Stanford Research Inst., Menlo Park, Calif., April 1966.

[24] [Brain '66[2]] "Graphical-data-processing research study and experimental investigation," A. E. Brain, P. E. Hart, and J. H. Munson, Tech. Rept. ECOM-01901-25, Stanford Research Inst., Menlo Park, Calif., December 1966.

[25] [Braverman '62] "Experiments on machine learning to recognize visual patterns," E. M. Braverman, translated from *Avtomat. i Telemekh.*, vol. 23, pp. 349–364, March 1962.

[26] [Braverman '66[1]] "Estimation of the rate of convergence of algorithms based on the potential functions," E. M. Braverman and B. Pyatnitskii, *Automation and Remote Control*, vol. 27, pp. 80–101, January 1966.

[27] [Braverman '66[2]] "The method of potential functions in the problem of training machines to recognize patterns without a trainer," E. M. Braverman, *Automation and Remote Control*, vol. 27, pp. 1748–1771, October 1966.

[28] [Bryan '63] "Experiments in adaptive pattern recognition," J. S. Bryan, *Proc. 1963 Bionics Symp. (Preprint)*, USAF.

[29] [Casey '65[1]] "Linear reduction of dimensionality in pattern recognition," R. G. Casey, Ph.D. dissertation, Dept. of Elec. Engrg., Columbia University, New York, N. Y., 1965; and Research Rept. RC-1431, IBM Corp., Yorktown Heights, N. Y., March 19, 1965.

[30] [Casey '65[2]] "An experimental comparison of several design algorithms used in pattern recognition," R. G. Casey, Rept. RC 1500, Watson Research Center, IBM Corp., Yorktown Heights, N. Y., November 1965.

[31] [Casey '66] "Recognition of printed Chinese characters," R. G. Casey and G. Nagy, *IEEE Trans. Electronic Computers*, vol. EC-15, pp. 91–101, February 1966.

[32] [Casey '67[1]] "An autonomous reading machine," R. G. Casey and G. Nagy, *IEEE Trans. Electronic Computers*, vol. C-17, May 1968; also Research Rept. RC-1768, IBM Corp., Yorktown Heights, N. Y., February 1967.

[33] [Casey '67[2]] "Normalization of hand-printed characters," R. G. Casey, Research Rept., Watson Research Center, IBM Corp., Yorktown Heights, N. Y., in press. See also "Normalization and recognition of two-dimensional patterns with linear distortion by moments," K. Udagawa *et al.*, *Electronics and Commun. Japan*, vol. 47, pp. 34–35, June 1964.

[34] [Chien '66] "A modified sequential recognition machine using time-varying stopping boundaries," Y. T. Chien and K. S. Fu, *IEEE Trans. Information Theory*, vol. IT-12, pp. 206–214, April 1966.

[35] [Chow '57] "An optimum character recognition system using decision functions," C. K. Chow, *IRE Trans. Electronic Computers*, vol. EC-6, pp. 247–254, December 1957.

[36] [Chow '62] "A recognition method using neighbor dependence," C. K. Chow, *IRE Trans. Electronic Computers*, vol. EC-11, pp. 683–690, October 1962.

[37] [Chow '65] "Statistical independence and threshold functions," C. K. Chow, *IEEE Trans. Electronic Computers* (*Short Notes*), vol. EC-14, pp. 66–68, February 1965.

[38] [Chow '66] "A class of nonlinear recognition procedures," C. K. Chow, *IEEE Trans. Systems Science and Cybernetics*, vol. SSC-2, pp. 101–109, December 1966.

[39] [Clemens '65] "Optical character recognition for reading machine applications," J. C. Clemens, Ph.D. dissertation, Dept. of Elec. Engrg., M.I.T., Cambridge, Mass., September 1965.

[40] [Cooley '65] "An algorithm for the machine calculation of complex Fourier series," J. W. Cooley and J. W. Tukey, *Math. Comput.*, vol. 19, no. 90, pp. 297–301, 1965.

[41] [Cooper '64] "Nonsupervised adaptive signal detection and pattern recognition," D. B. Cooper and P. W. Cooper, *Information and Control*, vol. 7, pp. 416–444, 1964.

[42] [Cooper '64] "Hyperplanes, hyperspheres, and hyperquadrics as decision boundaries," P. W. Cooper, in *Computer and Information Science*. Washington, D. C.: Spartan, 1964, pp. 111–139.

[43] [Cover '66] "The nearest neighbor decision rule," T. M. Cover and P. Hart, presented at the Internat'l Symp. on Decision Theory, 1966; also "Nearest neighbor pattern classification," T. M. Cover and P. E. Hart, *IEEE Trans. Information Theory*, vol. IT-13, pp. 21–27, January 1967.

[44] [Dorofeyuk '66] "Teaching algorithm for a pattern recognition machine without a teacher, based on the method of potential functions," A. A. Dorofeyuk, *Automation and Remote Control*, vol. 27, pp. 1728–1737, October 1966.

[45] [Doyle '60] "Recognition of sloppy, hand-printed characters," W. Doyle, *Proc. Western Joint Computer Conf.*, pp. 133–142, May 1960.

[46] [Duda '64] "Training a threshold logic unit with imperfectly classified patterns," R. O. Duda and R. C. Singleton, presented at the Western Joint Computer Conf., Los Angeles, Calif., August 1964.

[47] [Duda '66] "Pattern classification by iteratively determined linear and piecewise linear discriminant functions," R. O. Duda and H. Fossum, *IEEE Trans. Electronic Computers*, vol. EC-15, pp. 221–232, April 1966.

[48] [Efron '63] "The perception correction procedure in non-separable situations," B. Efron, Applied Physics Lab. Research Note, Stanford Research Inst., Menlo Park, Calif., August 1963.

[49] [Estes '65] "Measurement selection for linear discriminants used in pattern classification," S. E. Estes, Ph.D. dissertation, Stanford University, Stanford, Calif., 1964; also Research Rept. RJ-331, IBM Corp., San Jose, Calif., April 1, 1965.

[50] [Flanagan '65] *Speech Analysis Synthesis and Perception*, J. L. Flanagan. New York: Academic Press, 1965.

[51] [Fortier '65] "Clustering procedures," J. J. Fortier and H. Solomon, presented at the Internat'l Symp. on Multivariate Analysis, Dayton, Ohio, June 1965; also Rept. 7, Dept. of Statistics, Stanford University, Stanford, Calif., March 20, 1964.

[52] [Fralick '64] "The synthesis of machines which learn without a teacher," S. C. Fralick, Tech. Rept. 6103-8, Stanford Electronics Lab., Stanford, Calif., April 1964; also "Learning to recognize patterns without a teacher," *IEEE Trans. Information Theory*, vol. IT-13, pp. 57–65, January 1967.

[53] [Freeman '62] "On the digital computer classification of geometric line patterns," by H. Freeman, *Proc. Nat'l Electronics Conf.*, vol. 18, pp. 312–314, October 1962.

[54] [Friedman '66] "On some invariant criteria for grouping data," H. P. Friedman and J. Rubin, presented at the Biometric Session on Cluster Analysis at the Statistical Meetings, Brookhaven, N. Y., April 1966; also Rept. 39.001, IBM New York Scientific Center, April 1966.

[55] [Fu '64] "A sequential decision approach to problems in pattern recognition and learning," K. S. Fu and C. H. Chen, presented at the 3rd Symp. on Discrete Adaptive Processes, Chicago, Ill., October 1964.

[56] [Gaston '63] "A simple test for linear separability," C. A. Gaston, *IEEE Trans. Electronic Computers* (*Correspondence*), vol. EC-12, pp. 134–135, April 1963.

[57] [Glanz '65] "Statistical extrapolation in certain adaptive pattern recognition systems," F. H. Glanz, Tech. Rept. 6767-1, Stanford Electronics Lab., Stanford, Calif., May 1965.

[58] [Gold '59] "Machine recognition of hand-sent Morse code," B. Gold, *IRE Trans. Information Theory*, vol. IT-5, pp. 17–24, March 1959.

[59] [Good '65] *The Estimation of Probabilities*, I. J. Good, Research Monograph 30. Cambridge, Mass.: M.I.T. Press, 1965.

[60] [Greanias '63] "The recognition of handwritten numerals by contour analysis," E. C. Greanias, P. F. Meagher, R. J. Norman, and P. Essinger, *IBM J. Research and Develop.*, vol. 7, pp. 14–22, January 1963.

[61] [Griffin '62] "An optical character recognition system using a Vidicon scanner," E. Griffin, in *Optical Character Recognition*. Washington, D. C.: Spartan, 1962, pp. 73–84.

[62] [Griffin '63] "A pattern identification system using linear decision functions," J. S. Griffin, J. H. King, and C. J. Tunis, *IBM Sys. J.*, vol. 2, pp. 248–267, December 1963.

[63] [Groner '64] "Statistical analysis of adaptive linear classifiers," G. F. Groner, Tech. Rept. 6761, Stanford Electronics Lab., Stanford, Calif., April 1964.

[64] [Groner '66] "Real-time recognition of handprinted symbols," G. F. Groner, presented at the IEEE Pattern Recognition Workshop, Puerto Rico, October 1966.

[65] [Hawkins '66] "Automatic shape detection for programmed terrain classification," J. K. Hawkins, G. T. Elerding, K. W. Bixby, and P. A. Haworth, *Proc. Soc. Photographic Instrumentation Engrs.* (Boston, Mass.), June 1966.

[66] [Hennis '66] "The recognition of unnurtured characters in a multifont application," R. Hennis, presented at the IEEE Pattern Recognition Workshop, Puerto Rico, October 1966.

[67] [Hestenes '52] "Method of conjugate gradients for solving linear systems," M. R. Hestenes and E. Stiefel, *J. Research NBS*, vol. 49, pp. 409–436, 1952.

[68] [Highleyman '62] "Linear decision functions, with application to pattern recognition," W. H. Highleyman, *Proc. IRE*, vol. 50, pp. 1501–1514, June 1962.

[69] [Highleyman '63] "Data for character recognition studies," W. H. Highleyman, *IEEE Trans. Electronic Computers* (*Correspondence*), vol. EC-12, pp. 135–136, April 1963.

[70] [Ho '65] "An algorithm for linear inequalities and its applications," Y.-C. Ho and R. L. Kashyap, *IEEE Trans. Electronic Computers*, vol. EC-14, pp. 683–688, October 1965.

[71] [Hoff '62] "Learning phenomena in networks of adaptive switching circuits," M. E. Hoff, Jr., Tech. Rept. 1554-1, Stanford Electronics Lab., Stanford, Calif., July 1962.

[72] [Horwitz '61] "Pattern recognition using autocorrelation," L. P. Horwitz and G. L. Shelton, Jr., *Proc. IRE*, vol. 49, pp. 175–185, January 1961.

[73] [Hu '62] "Visual pattern recognition by moment invariants," M.-K. Hu, *IRE Trans. Information Theory*, vol. IT-8, pp. 179–187, February 1962.

[74] [Hubel '62] "Receptive fields, binocular interaction and functional architecture in the cat's visual cortex," D. H. Hubel and T. N. Wiesel, *J. Physiol.*, vol. 160, pp. 106–154, 1962.

[75] [Ide '66] "An experimental investigation of an unsupervised adaptive algorithm," E. R. Ide and C. J. Tunis, Rept. TR 01.967, IBM Systems Development Div., Endicott, N. Y., July 29, 1966.

[76] [Joseph '60] "On predicting perceptron performance," R. D. Joseph, *IRE Nat'l Conv. Rec.*, pt. 2, 1960.

[77] [Kamentsky '63] "Computer automated design of multifont print recognition logic," L. A. Kamentsky and C. N. Liu, *IBM J. Research and Develop.*, vol. 7, no. 1, pp. 2–13, 1963.

[78] [Kamentsky '64] "A theoretical and experimental study of a model for pattern recognition," L. A. Kamentsky and C. N. Liu, in *Computer and Information Sciences*, Washington, D. C.: Spartan, 1964, pp. 194-218.

[79] [Kanal '62] "Evaluation of a class of pattern-recognition networks," L. Kanal, in *Biological Prototypes and Synthetic Systems*. New York: Plenum, 1962, pp. 261–270.

[80] [Kashyap '66] "Recovery of functions from noisy measurements taken at randomly selected points and its application to pattern classification," R. L. Kashyap and C. C. Blaydon, *Proc. IEEE (Letters)*, vol. 54, pp. 1127–1129, August 1966.

[81] [Kesler '61] "Analysis and simulations of a nerve cell model," C. Kesler, Cognitive Systems Research Program Rept. 2, Cornell University, Ithaca, N. Y., May 1961.

[82] [Kesler '63] "Further studies of reinforcement procedures and related problems," C. Kesler, in *Collected Technical Papers*, vol. 2, Cognitive Systems Research Program. Ithaca, N. Y.: Cornell University, July 1963, pp. 16–64.

[83] [Kiessling '65] "The generation of linearly separable codes for adaptive threshold networks," C. Kiessling, M.S. thesis, Dept of Elec. Engrg., Cornell University, Ithaca, N. Y., June 1965.

[84] [Koford '64] "Adaptive pattern dichotomization," J. S. Koford, Tech. Rept. 6201-1, Stanford Electronics Lab., Stanford, Calif., April 1964.

[85] [Konheim '64] "Linear and nonlinear methods in pattern classification," A. G. Konheim, *IBM J. Research and Develop.*, July 1964.

[86] [Kuhl '63] "Classification and recognition of hand-printed characters," F. Kuhl, *IEEE Internat'l Conv. Rec.*, pt. 4, pp. 75–93, March 1963.

[87] [Ledley '64] "Concept analysis by syntax processing," R. Ledley and J. Wilson, *Proc. Am. Documentation Inst.*, vol. 1, October 1964.

[88] [Liu '66] "An experimental investigation of a mixed-font print recognition system," C. N. Liu and G. L. Shelton, Jr., *IEEE Trans. Electronic Computers*, vol. EC-15, pp. 916–925, December 1966.

[89] [Liu '64] "A programmed algorithm for designing multifont character recognition logics," C. N. Liu, *IEEE Trans. Electronic Computers*, vol. EC-13, pp. 586–593, October 1964.

[90] [Low '63] "Influence of component imperfections on trainable systems," P. R. Low, Paper 6-3, presented at the WESCON Conv., 1963.

[91] [Lucky '65] "Automatic equalization for digital communication," R. W. Lucky, *Bell Sys. Tech. J.*, vol. 44, pp. 547–588, April 1965.

[92] [Lucky '66] "Techniques for adaptive equalization of digital communication systems," R. W. Lucky, *Bell Sys. Tech. J.*, vol. 45, pp. 255–286, February 1966.

[93] [MacQueen '67] "Some methods for classification and analysis of multivariate observations," J. MacQueen, *Proc. 5th Berkeley Symp. on Statistics and Probability*. Berkeley, Calif.: University of California Press, 1967, pp. 281–297.

[94] [Manning '66] "An application of statistical classification procedures to teleseismic event classification," J. E. Manning, M. C. Grignetti, and P. R. Miles, Rept. 1372, Bolt, Beranek, and Newman, Cambridge, Mass., July 1966.

[95] [Mays '64] "Effect of adaptation parameters on convergence time and tolerance for adaptive threshold elements," C. H. Mays, *IEEE Trans. Electronic Computers (Short Notes)*, vol. EC-13, pp. 465–468, August 1964.

[96] [McLaughlin '67] "*N*th order autocorrelations in pattern recognition," J. McLaughlin and J. Raviv, presented at the *IEEE Internat'l Symp. on Information Theory* (Athens, Greece), August 1967.

[97] [McCormick '63] "The Illinois pattern recognition computer—ILLIAC III," B. H. McCormick, *IEEE Trans. Electronic Computers*, vol. EC-12, pp. 791–813, December 1963.

[98] [Michener '57] "A quantitative approach to a problem in classification," C. D. Michener and R. R. Sokal, *Evolution*, vol. 11, pp. 130–162, June 1957.

[99] [Minsky '61] "Steps toward artificial intelligence," M. Minsky, *Proc. IRE*, pp. 8–30, January 1961.

[100] [Moore '65] "Statistical analysis and functional interpretation of neuronal spike data," G. P. Moore, D. H. Perkel, and J. P. Segundo, *Annual Rev. Physiol.*, vol. 28, pp. 493–522, 1965.

[101] [Munson '66] "The recognition of handprinted text," J. H. Munson, presented at the IEEE Pattern Recognition Workshop, Puerto Rico, October 1966.

[102] [Munson '67] "Experiments with Highleyman's data," J. H. Munson, R. O. Duda, and P. E. Hart, Research Rept., Stanford Research Institute, Menlo Park, Calif., October 1967 (submitted for publication to *IEEE Trans. Electronic Computers*).

[103] [Nagy '66] "Self-corrective character recognition system," G. Nagy and G. L. Shelton, Jr., *IEEE Trans. Information Theory*, vol. IT-12, pp. 215–222, April 1966.

[104] [Narasimhan '64] "Labeling schemata and syntactic description of pictures," R. Narasimhan, *Information and Control*, vol. 7, pp. 151–179, June 1964.

[105] [Nilsson '65] *Learning Machines*, N. J. Nilsson. New York: McGraw-Hill, 1965.

[106] [Ornstein '65] "Computer learning and the scientific method: A proposed solution to the information theoretical problem of meaning," L. Ornstein, *J. Mt. Sinai Hosp.*, vol. 32, pp. 437–494, July 1965.

[107] [Papert '66] "Properties of two-layered threshold nets," S. Papert and M. Minsky, presented at the IEEE Pattern Recognition Workshop, Puerto Rico, October 1966; also M.I.T. Monograph MAC-M-358, Cambridge, Mass., September 1967.

[108] [Pask '60] "The simulation of learning and decision-making behavior," G. Pask, in *Aspects of the Theory of Artificial Intelligence*. New York: Plenum, 1962, pp. 165–211.

[109] [Patrick '66] "Nonsupervised sequential classification and recognition of patterns," E. A. Patrick and J. C. Hancock, *IEEE Trans. Information Theory*, vol. IT-12, pp. 362–372, July 1966.

[110] [Penrose '55] "Generalized inverse for matrices," R. A. Penrose, *Proc. Cambridge Phil. Soc.*, vol. 51, pp. 406–413, 1955.

[111] [Perotto '63] "A new method for automatic character recognition," P. G. Perotto, *IEEE Trans. Electronic Computers*, vol. EC-13, pp. 521–526, October 1963.

[112] [Peterson '65] "Discriminant functions: Properties, classes, and computational techniques," D. W. Peterson, Tech. Rept. 6761-2, Stanford Electronics Lab., Stanford, Calif., April 1965.

[113] [Pfaltz '67] "Computer representation of planar regions by their skeletons," J. L. Pfaltz and A. Rosenfeld, *Commun. ACM*, vol. 10, pp. 119–123, February 1967.

[114] [Pless '65] "PEPR system (precise encoding and pattern recognition)," I. Pless, *IEEE Trans. Nuclear Science*, vol. NS-12, pp. 279–290, August 1965.

[115] [Prewitt '66] "Pictorial data processing methods in microscopy," J. M. Prewitt, B. H. Mayall, and M. L. Mendelsohn, *Proc. Soc. Photographic Instrumentation Engrs.* (Boston, Mass.), June 1966.

[116] [Rabinow '62] "Developments in character recognition machines at Rabinow Engineering Company," J. Rabinow, in *Optical Character Recognition*. Washington, D. C.: Spartan, 1962, pp. 27–50.

[117] [Raviv '65] "Linear methods for biological data processing," J. Raviv and D. N. Streeter, Research Rept. RC-1577, IBM Corp., Yorktown Heights, N. Y., December 1965.

[118] [Raviv '67] "Decision making in Markov chains applied to the problem of pattern recognition," J. Raviv, *IEEE Trans. Information Theory*, vol. IT-13, pp. 536–551, October 1967.

[119] [Ridgeway '62] "An adaptive logic system with generalizing properties," W. C. Ridgeway III, Tech. Rept. 1556-1, Stanford Electronics Lab., Stanford, Calif., April 1962.

[120] [Rogers '60] "A computer program for classifying plants," D. J. Rogers and T. T. Tanimoto, *Science*, vol. 132, October 21, 1960.

[121] [Rosen '66] "A pattern recognition experiment with near-optimum results," C. A. Rosen and D. J. Hall, *IEEE Trans. Electronic Computers (Correspondence)*, vol. EC-15, pp. 666–667, August 1966.

[122] [Rosen '65] "Pattern separation by convex programming," J. B. Rosen, *J. Math. Anal. and Application*, vol. 10, pp. 123–134, 1965.

[123] [Rosenblatt '57] "The perceptron—A perceiving and recognizing

automaton," F. Rosenblatt, Rept. 85-460-1, Cornell Aeronautical Lab., Ithaca, N. Y., January 1957.

[124] [Rosenblatt '62[1]] "A comparison of several perceptron models," F. Rosenblatt, in *Self-Organizing Systems*. Washington, D. C.: Spartan, 1962.

[125] [Rosenblatt '62[2]] *Principles of Neurodynamics*, F. Rosenblatt. Washington, D. C.: Spartan, 1962.

[126] [Rosenfeld '66] "Sequential operations in digital picture processing," A. Rosenfeld and J. L. Pfaltz, *J. ACM*, vol. 13, pp. 471–494, October 1966.

[127] [Rubin '66] "Optimal classification into groups: An approach for solving the taxonomy problem," J. Rubin, Rept. 39.014, IBM New York Scientific Center, December 1966.

[128] [Scudder '65] "Probability of error of some adaptive pattern recognition machines," H. J. Scudder, III, *IEEE Trans. Information Theory*, vol. IT-11, pp. 363–371, July 1965.

[129] [Sebestyen '62] *Decision-Making Processes in Pattern Recognition*, G. Sebestyen, ACM Monograph. New York: Macmillan, 1962.

[130] [Selfridge, '60] "Pattern recognition by machine," O. G. Selfridge and U. Neisser, *Scientific American*, pp. 60–68, August 1960.

[131] [Sheng '64] "A method for testing and realization of threshold functions," C. L. Sheng, *IEEE Trans. Electronic Computers*, vol. EC-13, pp. 232–239, June 1964.

[132] [Singleton '62] "A test for linear separability as applied to self-organizing machines," R. C. Singleton, in *Self-Organizing Systems*. Washington, D. C.: Spartan, 1962.

[133] [Smith '68] "Pattern classifier design by linear programming," F. W. Smith, *IEEE Trans. Computers*, vol. C-17, pp. 367–372, April 1968.

[134] [Sokai '63] "The automatic speech recognition system for conversational sound," T. Sokai and S. Doshita, *IEEE Trans. Electronic Computers*, vol. EC-12, pp. 835–846, December 1963.

[135] [Sokal '63] *Principles of Numerical Taxonomy*, R. R. Sokal and P. H. A. Sneath. San Francisco, Calif.: W. H. Freeman and Co., 1963.

[136] [Specht '66] "Generalization of polynomial discriminant functions for pattern recognition," Specht, presented at the IEEE Pattern Recognition Workshop, Puerto Rico, October 1966.

[137] [Spragins '66] "Learning without a teacher," J. Spragins, *IEEE Trans. Information Theory*, vol. IT-12, pp. 223–229, April 1966.

[138] [Steinbuch '62] "Learning matrices and their applications," K. Steinbuch and U. A. W. Piske, *IEEE Trans. Electronic Computers*, vol. EC-12, pp. 846–862, December 1963; also "Adaptive systems in pattern recognition," H. Kazmierczak and K. Steinbuch, *IEEE Trans. Electronic Computers*, vol. EC-12, pp. 822–835, December 1963.

[139] [Tappert '66] "A model of speech perception and production," C. C. Tappert, presented at the 18th Internat'l Congress of Psychology Symp. on the Perception of Speech and Speech Mechanism, Moscow, USSR, August 1–7, 1966.

[140] [Teager '65] "Multi-dimensional visual processing," H. Teager, presented at the M.I.T. Pattern Recognition Seminar, May 26, 1965.

[141] [Tsypkin '66] "Adaptation, training, and self-organization in automatic systems," Y. Z. Tsypkin, *Automation and Remote Control*, vol. 27, pp. 16–52, January 1966.

[142] [Ueseka '67] "The use of the statistical nature of language in machine recognition," Y. Ueseka and S. Kuroki, *Proc. 1967 Conf. on Speech Communication and Processing (Preprint)*, USAF Office of Aerospace Research, pp. 113–118.

[143] [Wadsworth '66] "PEPR—A developmental system for rapid processing of bubble chamber film," B. F. Wadsworth, *Proc. Soc. Photographic Instrumentation Engrs.* (Boston, Mass.), June 1966.

[144] [Warshall '62] "A theorem on Boolean matrices," S. Warshall, *J. ACM*, vol. 9, no. 1, pp. 11–13, 1962.

[145] [Widrow '63] "Practical applications for adaptive data-processing system," B. Widrow *et al.*, Paper 11-4, presented at the WESCON Conv., San Francisco, Calif., 1963.

[146] [Widrow '64] "Pattern-recognizing control systems," B. Widrow and F. W. Smith, *Computer and Information Sciences*. Washington, D. C.: Spartan, 1964, pp. 288–317.

[147] [Wilks '62] *Mathematical Statistics*, S. S. Wilks. New York: Wiley, 1962.

[148] [Winder '63] "Threshold logic in artificial intelligence," R. O. Winder, *Proc. IEEE Winter General Meeting, Sessions on Artificial Intelligence*, pp. 107–128, January 1963.

Contributors

Mohamed S. Abou-Seada (M'68) was born in Cairo, Egypt, on October 4, 1937. He received the B.S. degree in electrical engineering from Cairo University in 1959, and the M.Sc. degree in electrical engineering while engaged in research and teaching at Iowa State University, Ames, in 1967. Currently he is a Ph.D. candidate at Iowa State University.

In 1959 he was employed by the General Organization for Executing the Five-Year Industrial Plan (now the General Organization for Industrialization), Cairo, Egypt. In 1962 he was sent by the Egyptian Ministry of Industry to the Soviet Union to study the Suez Thermal Power Plant Project. From 1959 to 1964 he taught part-time at Cairo University. In 1964 he received a one-year scholarship from the Institute of International Education (IIE) to do graduate study in the United States.

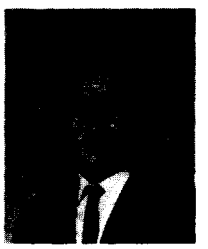
Mr. Abou-Seada is a member of Phi Kappa Phi and Sigma Xi.

Enval S. Barrekette was born in New York, N. Y., on February 18, 1931. He received the A.B. degree from Columbia College, New York, N. Y., in 1952, and the B.S. degree in civil engineering from Columbia University in 1953; he received the M.S. degree in flight structure and the Ph.D. degree in engineering mechanics from Columbia University in 1956 and 1959, respectively. From 1955 to 1957 he was a Guggenheim Fellow.

He was an Assistant Professor of Civil Engineering at Columbia University from 1959 to 1960, and has since been an Adjunct Associate Professor of Civil Engineering. Since 1960 he has been on the research staff at the IBM Watson Research Center, Yorktown Heights, N. Y., first as Technical Assistant to the Director of Research, then as Manager of Exploratory Systems, and currently as Manager of Electro-Optical Technologies.

Dr. Barrekette is a licensed professional engineer in New York, and a member of the American Institute of Aeronautics and Astronautics, the American Society of Civil Engineers, Phi Beta Kappa, Tau Beta Pi, and Sigma Xi.

❖



Chi-Tsong Chen (S'64–M'66) was born in Taiwan, China, on January 7, 1936. He received the B.S. degree in mechanical engineering from the National Taiwan University, China, in 1958, the M.S. degree in electrical engineering from the National Chiao-Tung University, China, in 1960, and the Ph.D. degree in electrical engineering from the University of California, Berkeley, in 1966.

From 1961 to 1962 he was a Lecturer at Taipei Institute of Technology, Taiwan. From

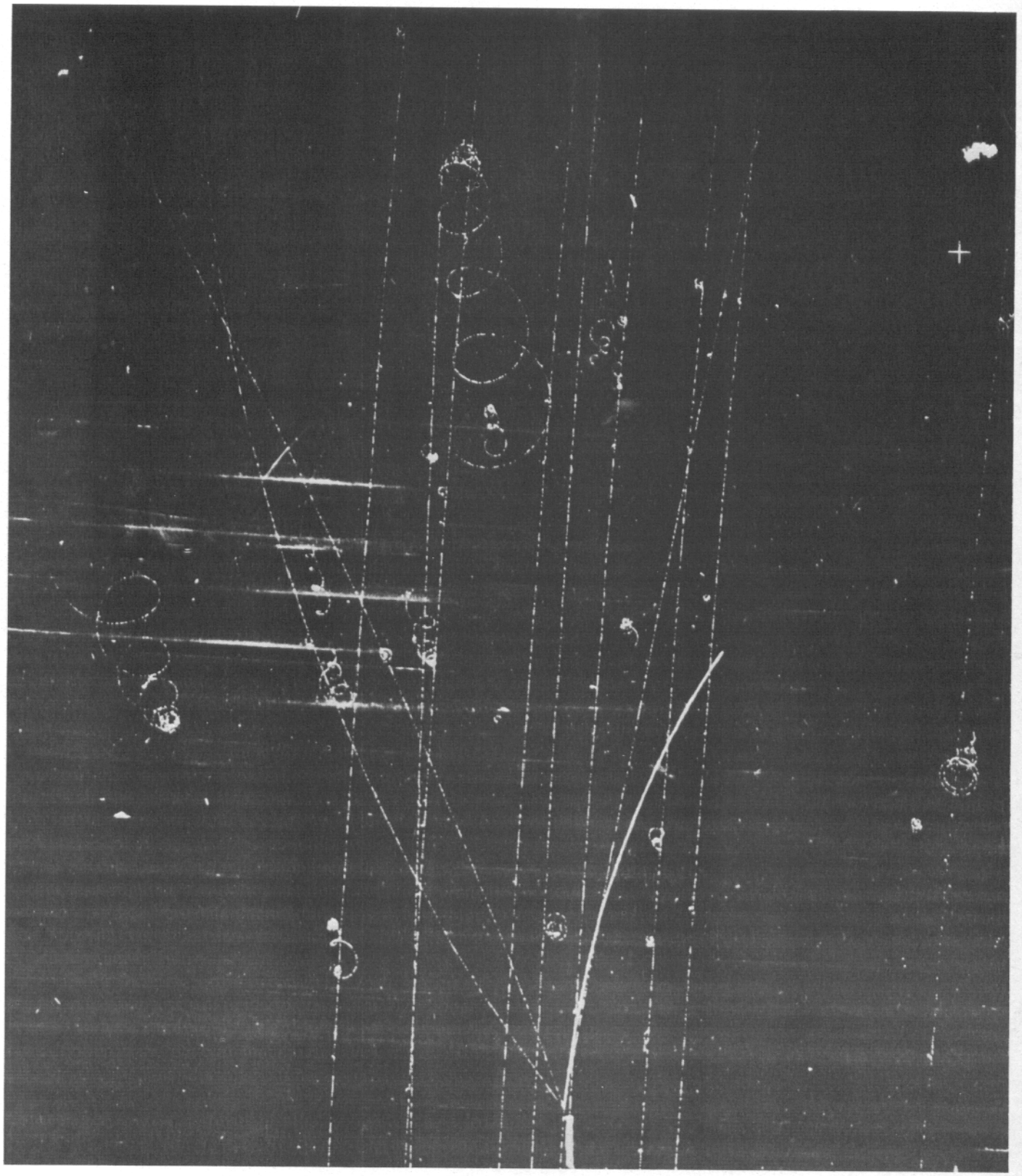


Fig. 16. A bubble chamber photograph showing particle tracks. The discontinuous nature of the tracks and the noisy background is inherent in the process. Two or three simultaneous exposures are generally obtained, for three-dimensional localization of the tracks. Note the fiducial marks. The photograph was obtained through the courtesy of the Stanford Linear Accelerator Center.