Dynamic Speaker Adaptation for Feature-Based Isolated Word Recognition

RICHARD M. STERN, MEMBER, IEEE, AND MOSHÉ J. LASRY, MEMBER, IEEE

Abstract—**In this paper, we describe efforts to improve the performance of FEATURE, the Carnegie-Mellon University speaker-independent speech recognition system that classifies isolated letters of the English alphabet by enabling the system to learn the acoustical characteristics of individual speakers. Even when features are designed to be speaker-independent, it is frequently observed that feature values may vary more from speaker to speaker for a single letter than they vary from letter to letter. In these cases, it is necessary to adjust the system's statistical description of the features of individual speakers to obtain improved recognition performance. This paper describes a set of dynamic adaptation procedures for updating expected feature values during recognition. The algorithm uses maximum *a posteriori* probability (MAP) estimation techniques to update the mean vectors of sets of feature values on a speaker-by-speaker basis. The MAP estimation algorithm makes use of both knowledge of the observations input to the system from an individual speaker and the relative variability of the features' means within and across all speakers. In addition, knowledge of the covariance of the features' mean vectors across the various letters enables the system to adapt its representation of similar-sounding letters after any one of them is presented to the classifier. The use of dynamic speaker adaptation improves classification performance of FEATURE by 49 percent after four presentations of the alphabet, when the system is provided with supervised training indicating which specific utterance had been presented to the classifier from a particular user. Performance can be improved by as much as 31 percent when the system is allowed to adapt passively in an unsupervised learning mode, without any information from individual users.**

I. INTRODUCTION

HUMAN beings perform remarkably well in recognizing and understanding the utterances of an unfamiliar speaker. While one occasionally requires a few seconds to become acquainted with the acoustical characteristics of new speakers with heavy foreign accents, subsequent human speech recognition performance is usually comparable to that obtained from a familiar talker. Most automatic speech recognition systems, on the other hand, fare much worse with unfamiliar talkers, particularly when these speakers pronounce words in an idiosyncratic fashion.

Manuscript received June 8, 1985; revised June 19, 1986. This work was supported in part by the National Science Foundation under Grant IRI-8512695 and in part by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 5167, monitored by the Air Force Avionics Laboratory under Contract N00039-85-C-0163. The views and conclusions contained in this paper are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the U.S. Government.

R. M. Stern is with the Department of Electrical and Computer Engineering and the Department of Computer Science, Carnegie-Mellon University, Pittsburgh, PA 15213.

M. J. Lasry is with Calltalk Ltd., Tel-Aviv 67214, Israel.

IEEE Log Number 8613862.

The general goal of speaker adaptation in a speech recognition system is to cause the system to learn the acoustical characteristics of individual speakers. In template-based systems, this is generally accomplished by making the reference templates speaker-specific in some fashion. For example, performance of the Harpy system was substantially improved by replacing the initial speaker-independent reference templates for each phone by the average of templates derived from the current user [1]. Zelinski and Class [2] describe a learning procedure in which reference templates are obtained from a running average of time-normalized representations of the utterances. Finally, Grenier and his colleagues [3], [4] have described a procedure that performs speaker-to-speaker transformations of the reference templates used in a template-matching system.

If the speaker-to-speaker variability of the reference templates and the variability of the observed data can both be characterized statistically, Bayesian learning procedures can be applied to optimally combine *a priori* speaker-independent knowledge with the speaker-specific knowledge obtained from actual observations from that speaker. For example, Brown *et al.* [5] have used Bayesian estimation procedures to adjust the statistics characterizing the acoustical observation vectors on a speaker-by-speaker basis in applying dynamic adaptation to a continuous digit recognition system based on hidden Markov modeling. In systems that perform statistical classifications on the basis of measured feature values, speaker adaptation can be achieved by modifying the parameters used by the system to classify an utterance into one of the possible classes.

Much recent work at Carnegie-Mellon University has been devoted to the development of speech recognition systems that exploit features of acoustically confusable utterances that can be used to perform fine phonetic distinctions in classifying words on a speaker-independent basis. One result of our research has been the recognition system called FEATURE that classifies the letters of the English alphabet based on the observed values of about 60 acoustical features [6], [7].

In this paper, we describe and discuss methods by which the CMU feature-based isolated letter recognition system FEATURE can adapt to the acoustical characteristics of individual speakers. FEATURE [6] performs a series of measurements on an input sound and uses these feature

values to classify the sound using statistical pattern recognition techniques. Even though the features used in the system were designed to be speaker-independent, the variation of individual feature values for a given letter is frequently smaller for a given speaker than the variation of values of that feature over all speakers. For example, across all speakers a given feature may have an expected value of 5 for **M** and 10 for **N**, but a specific individual speaker may produce average values of 9 for the particular feature for the letter **M** and 14 for **N**. In such cases, there may be significant overlap of the feature distributions for sets of letters over all speakers, and the performance of the recognition system can be greatly improved by adapting the statistical characterization of the feature values to the individual speaker.

We have chosen to apply maximum *a posteriori* probability (MAP) estimation techniques to dynamically update the mean vectors characterizing the random feature values on a speaker-by-speaker basis as new utterances are presented to the classifier. This approach to speaker adaptation was motivated by the informal observation that for many features the major speaker-dependent effect is in fact a shift in the mean value of the features for a particular letter. In general, MAP estimation specifies an optimal combination of *a priori* information about statistical parameters (such as the mean and covariance of a random vector) with the collection of observations input thus far to the classifier for a particular speaker. The classical MAP estimation technique has been previously applied to many types of statistical problems, including estimation of vocal tract length [8]. It is also possible, of course, to estimate the covariance matrices of the feature values on a speaker-by-speaker basis as well as their mean values. However, accurate estimation of these covariance matrices requires a larger database of utterances than is presently available, and the process of estimating covariance matrices for individual speakers would greatly increase the computational complexity required for the adaptation process. For these reasons, we have not attempted to estimate covariance matrices for individual speakers in the present study.

An important and novel attribute of our implementation of the MAP estimation algorithm is that the mean vectors of the features are characterized as random variables that may be correlated from letter to letter. This approach enables the system to use observed samples of the letter **M**, for example, to update estimates of the expected feature values for the letter **N** as well as **M**. As a result, the system adapts more rapidly to individual speakers than it would if all the mean values of the features were assumed to be statistically independent from letter to letter, but at the cost of greater computational complexity and storage requirements.

In Section II of this paper, we briefly review the structure of the CMU feature-based recognition system FEATURE and the procedures by which the mean values of the features can be dynamically adapted or "tuned" to individual speakers. In Section III, we compare the performance of various implementations of the tuning algorithms in recognizing the English alphabet. The tuning procedures discussed in Sections II and III are forms of "supervised learning" in that the system must have knowledge of what utterance is actually spoken to perform the adaptation. We discuss in Section IV extensions of these algorithms that enable the system to adapt to new speakers without feedback from the user. Finally, we summarize the results of our work in Section V.

II. Classification and Adaptation Procedures

A. Feature-Based Letter Recognition

The CMU feature-based isolated letter recognition system FEATURE has been described in detail in previous publications [6], [7]. Briefly, FEATURE performs spectral and temporal analyses on each incoming utterance, producing about 60 measurements describing various attributes of the sound. These features include characterizations of formant information, energy contours of nonvoiced segments, and temporal information such as the time from the beginning of an utterance to the vowel onset. We have assumed that these feature values are random variables with jointly Gaussian probability density functions. While the multivariate Gaussian assumption is definitely incorrect for at least some of the features, it was adopted as the most efficient way to express information about covariances among the various feature values, which we have found to be important for many of the classification tasks.

Although the optimal Bayesian classifier is well specified if the probability densities of the features are known for each letter, we do not implement a classifier that directly chooses 1 of 26 classes based on the observed values of the approximately 60 features for several practical reasons. Specifically, some of the features were designed to be meaningful for only a subset of the letters, and these features do not produce Gaussian probability densities when other letters are presented to the system. Also, our database does not have a sufficient number of utterances to estimate accurately the covariance matrices needed to train such a classifier. For these and other reasons, we have used a decision-tree structure to perform the classifications. At each node of the decision tree, the utterance is classified into a small number of decision categories, based on a relatively small number of features that are relevant to the classification in question. A diagram of the decision tree that was used for most of the classifications of the complete alphabet in the present paper is shown in Fig. 1. While the features used at the various nodes generally differ, some of the same features are used at different nodes. At some nodes, classifications in more than one decision category are directed to the same node at the next lower level of the decision tree as is seen (for example) with nodes 1 and 2.

In analyzing dynamic speaker adaptation in FEATURE, we performed classifications directly on the basis of the observed feature values, rather than the linear trans-

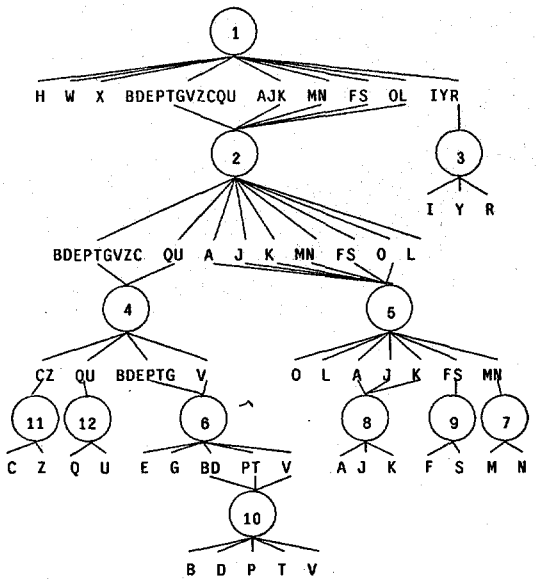


Fig. 1. Decision tree used to perform classifications of the alphabet. Note that some decision categories include more than one letter and that some categories are merged in subsequent nodes.

formations of these features used to obtain many of the results in [7]. At each level of the decision tree, further classifications were performed only on the nodes below the decision branch having greatest *a posteriori* probability, so errors made at any stage of the classification process are generally irrecoverable. Recoverable errors occur when a letter is mistakenly classified as a different letter of a class that is directed to the same lower-level node (such as when a **B** is classified as an **A** in node 1).

B. Dynamic Speaker Adaptation for the Letter Recognition System

In deriving the estimator to update the system's characterization of the mean vectors of the feature values for a given speaker, we model these mean vectors as random parameters that are fixed for a given speaker but vary from speaker to speaker according to a jointly Gaussian probability distribution. As noted above, we specifically assume that these mean values may be correlated from letter to letter as well as from feature to feature. Given the probability density functions (pdf's) for the observations and the mean values, the MAP estimate for the mean vector **M** is chosen to maximize the *a posteriori* probability

$$p(\mathbf{M}|\mathbf{x}) = \frac{p(\mathbf{x}|\mathbf{M}) \cdot p(\mathbf{M})}{p(\mathbf{x})}$$

where $p(\mathbf{x}|\mathbf{M})$ is the condition pdf for the observed feature vectors given their mean vectors, and $p(\mathbf{M})$ and $p(\mathbf{x})$ are the *a priori* pdf's for the mean vector and the observed features, respectively.

Except for the use of correlations of means from class to class, this general approach is well known as Bayesian learning of the mean of a probability density and is well documented (cf. [9]).

C. A Simple One-Dimensional Example

We can illustrate these calculations with a simple example. Assume that there are only two decision classes, C_1 and C_2, and that only a single feature is relevant to the decision. Let the set $\{x_i\}$ represent the N_1 observations that have already been presented to the classifier from samples of class C_1, and let $\{y_j\}$ represent the set of N_2 observations belonging to class C_2. We assume that the sets $\{x_i\}$ and $\{y_j\}$ are statistically independent jointly Gaussian random variables with means M_1 and M_2, respectively, and with variance σ_1^2 for class 1 and σ_2^2 for class 2. We further assume that these mean values M_1 and M_2 are correlated Gaussian random variables with means M_{01} and M_{02}. Finally, we assume that the variance of M_1 is ζ_1^2 and the variance of M_2 is ζ_2^2, and that the covariance of M_1 and M_2, defined as $E[(M_1 - M_{01})(M_2 - M_{02})]$, is equal to $\rho\zeta_1\zeta_2$.

Under the above assumptions, $\hat{M}_1$, the MAP estimate for M_1, is given by [10]

$$\hat{M}_1 = M_{01} + \alpha\left(\left(\frac{1}{N_1}\sum_{i=1}^{N_1} x_i\right) - M_{01}\right) + \beta\left(\left(\frac{1}{N_2}\sum_{j=1}^{N_2} y_j\right) - M_{02}\right) \tag{1a}$$

or

$$\hat{M}_1 = \alpha\left(\frac{1}{N_1}\sum_{i=1}^{N_1} x_i\right) + (1 - \alpha)M_{01} + \beta\left(\left(\frac{1}{N_2}\sum_{j=1}^{N_2} y_j\right) - M_{02}\right) \tag{1b}$$

where

$$\alpha = \frac{N_1\zeta_1^2(\sigma_2^2 + N_2\zeta_2^2(1 - \rho^2))}{(\sigma_1^2 + N_1\zeta_1^2)(\sigma_2^2 + N_2\zeta_2^2) - N_1N_2\rho^2\zeta_1^2\zeta_2^2} \tag{1c}$$

$$\beta = \frac{N_2\rho\sigma_1^2\zeta_1\zeta_2}{(\sigma_1^2 + N_1\zeta_1^2)(\sigma_2^2 + N_2\zeta_2^2) - N_1N_2\rho^2\zeta_1^2\zeta_2^2}. \tag{1d}$$

It is easy to verify that the first term of (1b) represents the contribution of the actual observations to the estimate, the second term represents the contribution of knowledge of the *a priori* mean for class C_1, and the third term represents the contribution of the data from the other class, C_2. It is also obvious that as N_1 increases, the first term (representing the observed feature values) will eventually predominate in the estimate. For a given N_1 and N_2, the relative salience of the first two terms is determined by the ratio of σ_1/ζ_1. For N_1 equal to 0, α equals 0 and the first term of $\hat{M}_1$ (the contribution of the observations from class 1) is equal to 0, as could be expected. When N_1 becomes very large, α tends to 1 and β tends to 0, so that the contribution of the second and third terms of $\hat{M}_1$ becomes negligible. Finally, if ρ equals 0, β is equal to 0 and there is no contribution of the observations from class 2: in that case, the random variables M_1 and M_2 are statistically independent. β is a monotonically increasing

function of $|\rho|$ so that the contribution of the observations from class 2 increases with the amount of correlation of M_1 and M_2.

As we obtain a larger number of labeled observations for a given class, we not only obtain a more accurate estimate of the mean value, but the variability of the estimate progressively decreases as well. Specifically, it can be shown that [10]

$$\zeta^2_{1N_1N_2} = E[\hat{M}_1 - M_1]^2 = \frac{\sigma_1^2\zeta_1^2(\sigma_2^2 + N_2\zeta_2^2(1-\rho^2))}{(\sigma_1^2 + N_1\zeta_1^2)(\sigma_2^2 + N_2\zeta_2^2) - N_1N_2\rho^2\zeta_1^2\zeta_2^2} \quad (2)$$

where $\zeta^2_{1N_1N_2}$ represents the variance of $\hat{M}_1$ after N_1 presentations of the $\{x_i\}$ and N_2 presentations of the $\{y_j\}$, and σ_1^2, σ_2^2, ζ_1^2, ζ_2^2, N_1, N_2, and ρ are as defined above. We note that if N_1 and N_2 equal 0, $\zeta^2_{1N_1N_2}$ equals ζ_1^2, and that $\zeta^2_{1N_1N_2}$ approaches 0 as N_1 goes to infinity.

As the estimate of the mean value of M_1 becomes increasingly reliable, the variance of the *a posteriori* pdf characterizing the *observations* $\{x_i\}$ decreases. Specifically, after N_1 observations of class 1 and N_2 observations of class 2, the next observation from class 1 is Gaussian with mean $\hat{M}$ and variance $\zeta^2_{1N_1N_2} + \sigma_1^2$. These statistics (and similar ones for class 2) are used by the classifier in making its decisions.

D. MAP Estimation with Correlated Decision Classes

In the general case of tuning the FEATURE system to new speakers, there are more than two decision classes and many features characterizing each decision class. Since we wish to update the estimates of all of the mean feature values simultaneously while considering correlations of mean values across the decision classes, it is more convenient to adopt a matrix formulation of the MAP estimation problem. We also introduce some special notational conventions to characterize the feature means and covariances over all decision classes, and the covariance of the mean values of the features from speaker to speaker, with concatenated vectors and matrices. Specifically, at each node of the decision tree of the FEATURE system, we adopt the following notation.

- C refers to the number of decision classes, or possible outcomes, of the decision node. Each decision class could be a single letter or a group of letters.
- D refers to the number of features used to classify the utterance at the node in question.
- $\mathbf{M}_1, \mathbf{M}_2, \cdots, \mathbf{M}_C$ are the D-dimensional mean vectors of class 1, 2, $\cdots$, C at the node. For each individual speaker, the $\{\mathbf{M}_k\}$ are random parameters.
- $\mathbf{M} = (\mathbf{M}_1^T, \mathbf{M}_2^T, \cdots, \mathbf{M}_C^T)^T$ is called the extended mean vector. It is a composite of C vectors of size D and therefore can be viewed as a vector of size CD.
- $\mathbf{M}_{01}, \mathbf{M}_{02}, \cdots, \mathbf{M}_{0C}$ are the *a priori* mean values over all speakers of the D-dimensional mean vectors of classes 1, 2, $\cdots$, C.
- $\mathbf{M}_0 = (\mathbf{M}_{01}^T, \mathbf{M}_{02}^T, \cdots, \mathbf{M}_{0C}^T)^T$ is called the extended *a priori* mean vector and is also of size CD.
- $\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_C$ are the D-dimensional sample mean values of the observed feature vectors of class 1, 2, $\cdots$, C at the node.
- $\mathbf{A} = (\mathbf{A}_1^T, \mathbf{A}_2^T, \cdots, \mathbf{A}_C^T)^T$ is called the extended sample mean and is also of size CD.
- $\mathbf{N}$ is a $CD \times CD$ diagonal matrix consisting of C diagonal blocks of size $D \times D$. The kth such block of $\mathbf{N}$ is equal to $N_k\mathbf{I}_D$ where $\mathbf{I}_D$ is the identity matrix of size D, and N_k is the number of observations of the kth class input thus far to the classifier.
- $\mathbf{S}_k$ is the $D \times D$ covariance matrix of the features of the kth decision class.
- $\mathbf{S}$ is the extended covariance matrix of the features. It is a block diagonal matrix of size $CD \times CD$, and the kth block is the covariance matrix $\mathbf{S}_k$.
- $\mathbf{Z}$, which is the covariance matrix of the extended mean vector, is referred to as the *mean cross covariance matrix*. It is equal to $E[(\mathbf{M} - \mathbf{M}_0)(\mathbf{M} - \mathbf{M}_0)^T]$ and is of size $CD \times CD$. The mean cross covariance matrix contains information about the correlations from class to class and speaker to speaker of the mean values of the features for a given speaker.

The mean cross covariance matrix $\mathbf{Z}$ can be intuitively understood as a composite of C^2 matrices, each of size $D \times D$. The (d_1, d_2)th element of the (c_1, c_2)th block is the covariance of the mean value of feature d_1 from class c_1 with the mean value of feature d_2 from class c_2. The structure of $\mathbf{Z}$ is illustrated in [10]. We assume, as before, that $\mathbf{M}$ is normally distributed about $\mathbf{M}_0$ with covariance matrix $\mathbf{Z}$. We also assume that the observation vector from the kth decision class is normally distributed about $\mathbf{M}_k$ with covariance matrix $\mathbf{S}_k$.

Using the assumptions stated above, and the additional (incorrect) assumption that the observations at each node of the decision tree are statistically independent, it can be shown [10] that at each node of the decision tree $\hat{\mathbf{M}}$, the exact MAP estimate of the extended mean vector is

$$\hat{\mathbf{M}} = \mathbf{Z}(\mathbf{Z} + \mathbf{N}^{-1}\mathbf{S})^{-1}\mathbf{A} + \mathbf{N}^{-1}\mathbf{S}(\mathbf{Z} + \mathbf{N}^{-1}\mathbf{S})^{-1}\mathbf{M}_0. \quad (3)$$

We note that the expression for $\hat{\mathbf{M}}$, which is similar in form to (1b), is a linear combination of the sample mean vector of the observations, $\mathbf{A}$, and the extended *a priori* mean vector $\mathbf{M}_0$.

Similarly, the extended covariance matrix characterizing the variability of $\hat{\mathbf{M}}$ after N_1, N_2, $\cdots$, $\mathbf{N}_c$ observations from the various classes is

$$E\left[(\hat{\mathbf{M}} - E[\hat{\mathbf{M}}])(\hat{\mathbf{M}} - E[\hat{\mathbf{M}}])^T\right] \equiv \mathbf{Z}_\mathbf{N} = \mathbf{Z}(\mathbf{S} + \mathbf{NZ})^{-1}\mathbf{S}. \quad (4)$$

The *a posteriori* pdf for observations for the kth decision class is assumed to be normally distributed about $\hat{\mathbf{M}}$ with a covariance matrix equal to the sum of $\mathbf{S}_k$ and the kth diagonal $D \times D$ block of $\mathbf{Z}_\mathbf{N}$.

E. Implementation of Extended MAP Estimation in FEATURE

As discussed above, FEATURE classifies the letters of the alphabet using a hierarchical decision tree. At most nodes of the tree, the system classifies a subset of the alphabet into a set of decision classes, each consisting of one or more letters. In the present evaluations of speaker adaptation, we used an implementation of FEATURE that made independent decisions at each node, and we implemented the MAP estimator of the mean values of the features separately from node to node.

It is not possible to implement the complete extended MAP estimation procedure described above at each node of the decision tree if the *a priori* statistics of the estimator are obtained from a limited set of training data. Specifically, the mean cross covariance matrix $\mathbf{Z}$ is nonsingular only if the number of speakers used to train the estimator is greater than CD, the product of the number of decision classes by the number of features used at a given node of the decision tree [11]. This is a rather restrictive condition. The implementation of FEATURE used in the present evaluation includes nodes with as many as 9 decision classes classified according to values of as many as 16 features, so the product CD can be as large as 144 for some nodes. Since the number of speakers used to train the classifier and estimator is as low as 19 for some of these evaluations, it was necessary to reduce the size of the mean cross covariance matrix. Our most frequent solution to this problem is to consider each feature in a given decision node separately and to construct a set of D mean cross covariance matrices of size $C \times C$, one matrix for each of the features. Since the maximum number of classes involved in a decision is always smaller than 19, the constraint of invertibility is always satisfied. This solution, which makes use of class-to-class correlations but ignores feature-to-feature correlations of the mean vectors, is referred to as **type C** tuning. Other methods have also been considered, and they are discussed below.

A second problem that complicates the evaluation of the system is the observation that $\mathbf{S}_k$, the actual class-conditional covariance matrix of the observed feature values, varies from speaker to speaker. Since our statistical model assumes that the covariance matrix $\mathbf{S}$ is speaker independent, we had to estimate its value by averaging the within-speaker covariance matrices over the set of speakers. Since only four utterances per letter per speaker are available, we do not know whether the speaker-to-speaker variation in $\mathbf{S}$ represents another fundamental attribute of speaker variability, or whether it is simply due to the lack of more training data from each speaker.

Once the system is "trained" by estimating $\mathbf{M}_0$, $\mathbf{S}$, and $\mathbf{Z}$, it is initialized for a new speaker by setting $\hat{\mathbf{M}}$ to $\mathbf{M}_0$, $\mathbf{Z}_N$ to $\mathbf{Z}$, and $\mathbf{N}$ to $\mathbf{0}$. After the system classifies each incoming letter, the user indicates which utterance has actually been spoken. Then, at each node of the decision tree, the system updates its estimate of $\hat{\mathbf{M}}$ according to (3), and it also computes a new value for $\mathbf{Z}_N$ according to (4), with the nodes considered independently of one another. Subsequent utterances are classified by assuming that the feature values are jointly Gaussian random variables with means obtained from $\hat{\mathbf{M}}$ and covariances from $\mathbf{Z}_N$ and $\mathbf{S}$.

III. Results—Tuning with Feedback

A. Evaluation Procedures

In this section, we compare the classification performance of FEATURE for the English alphabet, and some acoustically confusable subsets of the alphabet, with and without dynamic speaker adaptation. We also compare the performance of different implementations of the tuning algorithms. These evaluations were performed on a database of 20 speakers, 10 males and 10 females, using 4 utterances of each of the 26 English letters. The data in this section were obtained using a procedure in which the system was trained on 19 of the 20 speakers and tested on the 20th. This process was then repeated, using each of the 20 speakers as the "test" speaker in turn.

We noted in the previous section that the most general form of the MAP estimation algorithm usually cannot be used because the size of the mean cross covariance matrix is limited by the number of training speakers available. There are several ways to reduce the dimensionality of the mean cross covariance matrix by ignoring *a priori* class-to-class or feature-to-feature correlations in the mean values of the features for a given speaker. We consider in this section the performance of some of these (nominally suboptimal) implementations of the MAP learning algorithm. We identify these procedures by the following notational conventions.

- **Type CF** refers to the tuning procedure that makes use of both *class-to-class* and *feature-to-feature* correlation of the features' mean vectors for a given speaker. This is the most general form of the extended MAP estimation algorithm, and it is theoretically the optimal implementation for a given node of the decision tree, specified by (3) and (4).
- **Type C** refers to a tuning procedure that makes use of *class-to-class* correlations of the features' mean values, but ignores feature-to-feature correlations of these means.
- **Type F** refers to a procedure that makes use of *feature-to-feature* correlations, but not class-to-class correlations of the features' means. This is the classical MAP algorithm for learning the mean of a set of correlated random Gaussian vectors separately for each decision class.
- **Type I** refers to a tuning procedure that makes use of neither class-to-class nor feature-to-feature correlation. The features are updated completely *independently* of one another.
- **Type S** tuning refers to the simplest form of dynamic adaptation, which is obtained by setting the mean vector of the features equal to the *sample mean* of the observations. While this procedure is, strictly speaking, not a Bayesian estimator, it is included for comparison because of its simplicity and its use in previous studies (e.g., [1],

TABLE I
INFORMATION USED BY THE VARIOUS TUNING PROCEDURES

Type of Procedure	CF	F	C	I	S	0
Class-to-class correlations of means	*		*			
Feature-to-feature correlations of means	*	*				
A priori means	*	*	*	*		*
A priori covariances	*	*	*	*	*	*
A posteriori observations	*	*	*	*	*	

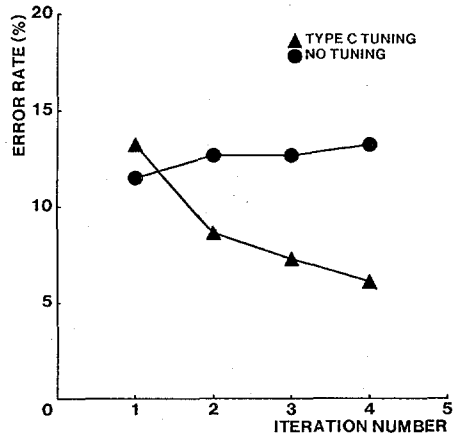


Fig. 2. Comparison of the error rate for the entire English alphabet using **type C** tuning versus no tuning, on an iteration-by-iteration basis.

[2]). **Type S** tuning uses a combination of the running sample covariance matrix of the mean vectors with the *a priori* covariance matrix of the data $\mathbf{S}_k$ to estimate the covariance of the observed data, so it can only be used after at least two utterances from each decision class have been presented to the classifier.

Table I summarizes these notational conventions. Each column represents a different adaptation procedure, and each row represents a different type of information. **Type 0** refers to no tuning. The asterisks indicate which type of information is used by each type of learning procedure.

B. Classification Results

Using the evaluation procedure described above, we compare in Fig. 2 the error rate for the entire alphabet using no tuning to the error rate with the **type C** (class-to-class correlation) tuning method. The results are averaged over the 20 speakers, and each successive iteration through the alphabet is plotted separately. We note that while the average untuned speaker-independent error rate remains at about 12.5 percent over the four sets of presentations, tuning to individual speakers with the **type C** procedure causes the error rate to steadily decrease to 6.2 percent by the fourth set of presentations. (The untuned error rate is slightly worse than that reported in [6] and [7] because we used a different implementation of the FEATURE system in our evaluation, without the linear discriminant analyses.)

In order to compare the contributions of various types of *a priori* knowledge toward reducing the classification error rate through tuning, we conducted more extensive performance comparisons for individual nodes that classify acoustically confusable subsets of letters. We selected for this further experimentation the letters **M** and **N**, and **B**, **D**, **P**, and **T** because the untuned error rates are relatively large for these sets of letters, and they can be substantially reduced by tuning to individual speakers. To reduce the dependence of the classification results on the statistical fluctuations of the relatively small number of data on hand for these evaluations, we averaged the classification performance with the four utterances of each letter represented in forward and reversed order. This causes the untuned error rate to appear as a symmetric function of the iteration number.

In Fig. 3 we compare the error rate for the letters **M** and **N** using the five different tuning procedures, **types CF, C, F, I,** and **S**. Performance was averaged over the 20 test speakers, as well as the forward and reversed order of presentation of the utterances, and each point represents a total of 80 classifications [(2 letters) * (2 presentation orders) * (20 speakers)]. Fig. 4 presents similar comparisons of performance in discriminating the letters **B, D, P,** and **T**, using **types C, F, I,** and **S** tuning. It is not possible to use **type CF** tuning to classify these four letters because the product of the number of letters and features exceeds the number of training speakers available. Each point in Fig. 4 represents a total of 160 classifications.

We note that the statistical significance of the data in Figs. 3 and 4 is limited by the small number of data available for classification in these first experiments, and using a simple Bernoulli model of the classification process and a nominal 10 percent error rate, the 95 percent confidence interval is about ± 6.5 percent for the data in Fig. 3 and ± 4.5 percent for the data in Fig. 4. Nevertheless, we wish to call attention to the following data trends, which we have observed in other similar comparisons as well.

- **Type CF** tuning produces better results than **type F** in the first iteration in Fig. 3, and **type C** tuning produces better performance than **type I** for the first iteration in Fig. 4. These results imply that the use of class-to-class correlations of the mean feature values can provide an improvement in performance.
- For the first iteration, **type F** performs worse than **type I**, and **type CF** performs worse than **type C**, implying that the use of feature-to-feature correlation can degrade performance. We believe that this apparently paradoxical result occurs because the actual feature-to-feature cross correlations of the mean feature values for these letters are small, and they are probably inaccurately estimated with limited training data.
- For these two sets of letters, the improvement in performance provided by the use of class-to-class correlation in tuning becomes negligible after three or four presentations of the stimulus set. After this point, the simple **type S** tuning based on the values of the sample means appears to perform as well or better, at least for these letters and features. This indicates that only a very small amount of speaker-dependent training is needed before the

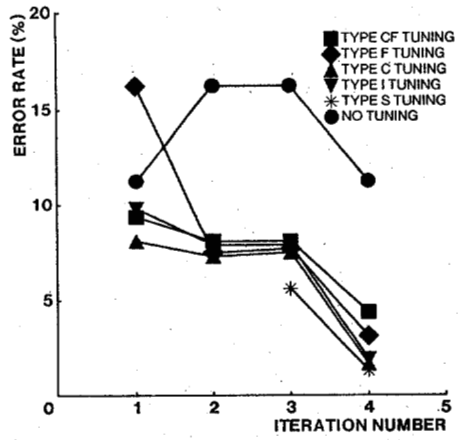


Fig. 3. Evolution of the error rate for the letters **M** and **N** for all types of tuning.

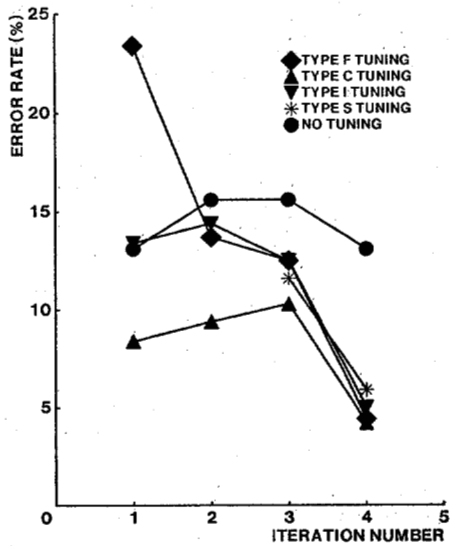


Fig. 4. Evolution of error rates for the letters **B, D, P,** and **T** for **types F, C, I,** and **S** tuning.

contribution of *a priori* information becomes of negligible value.

C. Discussion—Tuning with Feedback

Although the number of samples in these evaluations is limited, it appears that the use of any of our dynamic adaptation procedures improves the performance of the recognition system. Furthermore, the tuning procedures that make use of class-to-class correlations usually enable the system to adapt more rapidly to a new user than the methods that operate on a class-by-class basis, although further experimentation will need to be done with a larger database for this aspect of the system to be demonstrated more convincingly. On the other hand, the very good performance of the **type S** procedure after a few presentations of the test data indicates that there is a great deal of information contained in the observed feature values for an individual speaker, even without much speaker-independent information at all. The performance of the different types of tuning algorithms depends to some extent on the vocabulary as well as the set of features that are used. In classifying the entire English alphabet, we also found that the rate at which the tuning procedures decrease the error rate varies considerably from node to node in the decision tree.

It is difficult to quantitatively compare our results in speaker adaptation to those of other studies because the sets of experimental data to be classified differ from study to study. However, it is worthwhile to consider the types of information used in each study. Most template-based systems, including those described by Lowerre [1] and Zelinski and Class [2], make use only of averages of the observed data in constructing templates. In our feature-based classification system, this corresponds to the **type S** adaptation procedure. Brown *et al.* [5] used an optimal combination of this *a posteriori* information with *a priori* knowledge of the speaker-independent means and variances of template values. Their adaptation was performed independently for each feature and decision class, which corresponds to our **type I** adaptation procedure. Our more elaborate adaptation procedures (labeled **types C, F,** and **CF**) are similar in philosophy, but they can also make use of class-to-class and/or feature-to-feature correlations of the mean values if there are enough different speakers available to train the classifier. As seen in Figs. 3 and 4, the use of class-to-class correlation appears to be quite helpful in reducing the error rate of the data, while the feature-to-feature correlations of the mean feature values were not very helpful for the limited data that we examined. Brown *et al.* found (as did we) that the greatest advantage obtained using a Bayesian combination of *a priori* and *a posteriori* information (corresponding to our **type F** tuning) over using the *a posteriori* samples alone (corresponding to our **type S**) was observed when the amount of actual data available for a given speaker is relatively small. The system of Grenier [3], [4] differs in philosophy from the present work in that it adapts to new speakers using transition matrices that are precompiled and known for a given speaker. The acoustical characteristics of the current speaker are assumed to be unknown in the present work.

We noted in Section II-E that it is hard to determine whether the speaker-to-speaker variability we have observed for the covariance matrices of the feature values is intrinsic to the nature of the features or whether this variability is simply due to the very small size of our database. Since our set of training data contains only four samples of each letter for each of the speakers, we cannot rely on estimates of the covariance matrices on a speaker-by-speaker basis. Some studies (e.g., [12]) suggest that the number of samples needed to accurately evaluate a covariance matrix is at least five to ten times the size of the matrix, and this is far from the case for the training data for these experiments. If the covariance matrices of the features do indeed vary from speaker to speaker, one could use a procedure such as that described by Wishart (e.g., [13]) to dynamically update the covariance matrices as well as the mean vectors of each of the classes. Thus

far, we have not carefully explored the possibility of tuning covariance matrices to individual speakers.

IV. Tuning Without Feedback

While we have been encouraged by the improvements in recognition performance afforded by dynamic adaptation to individual speakers as described in the preceding section, it can be tedious for the user to inform the system which letter has actually been presented to the classifier after each utterance, and such user feedback may be unacceptable or impractical to implement in many applications. In this section, we discuss two extensions of the tuning procedures that do not require active participation on the part of the user. The first approach is referred to as confidence-based tuning and exploits the fact that the system makes fewer errors when the *a posteriori* probability of the chosen candidate letter is extremely high. The system then tunes by assuming that it has made a correct decision every time it classifies a sound with sufficiently high confidence. The second approach to unsupervised learning, called correlation-based tuning, exploits the letter-to-letter correlations of feature values for a given speaker. The system computes the running sample mean of a given feature over all utterances from all decision classes. The mean values of features for a given speaker are updated according to their *a priori* correlation with the average over all decision classes. We describe these two procedures in more detail and compare their performances for acoustically confusable subsets of the alphabet and for the alphabet as a whole.

A. Confidence-Based Tuning

Since FEATURE performs classifications by choosing the candidate letter with greatest *a posteriori* probability (given the observed feature vectors), it is possible to use that *a posteriori* probability as a measure of the confidence with which the decision has been made. For example, in Fig. 5 we show histograms of *a posteriori* probabilities from correct and incorrect classifications of the E-set letters (**B, C, D, E, G, P, T, V,** and **Z**). We note that while a percentage of misclassifications is made with high *a posteriori* probability, almost all of the decisions made with extremely high confidence are correct decisions.

In confidence-based tuning, we establish a threshold of *a posteriori* probability for the letters recognized by the classifier. If the actual *a posteriori* probability of a particular letter exceeds the threshold, the estimates of the means of the feature vectors (and the covariance matrices of the observed features) are updated according to the tuning procedures specified above, with the system assuming that the classification decision has been a correct one. If the *a posteriori* probability of a classified letter falls below that threshold, the classifier moves on to the next utterance without updating the means and covariances. Clearly, the performance of the system depends to some extent on the particular value of probability used as the threshold. Tuning will be implemented on a greater fraction of the utterances presented to the classifier as the threshold probability is decreased, but the percentage of times that the system will tune incorrectly will also increase. The system tunes after every utterance if the threshold *a posteriori* probability is 0.00; if the threshold is above 1.00, the system never tunes.

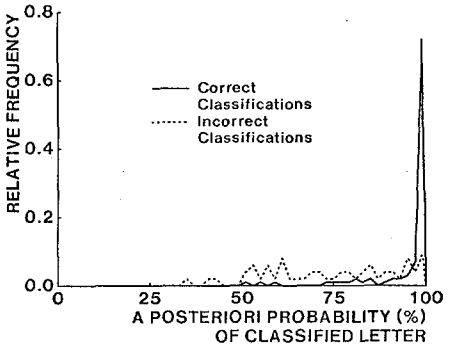


Fig. 5. Typical histogram of *a posteriori* probabilities for classifications of letters of the alphabet.

B. Correlation-Based Tuning

Correlation-based tuning adjusts the mean values of the features according to their correlation with the average mean values over all decision classes. This method is based on a particular application of the extended MAP learning procedure to a two-class, one-feature case, so we will discuss the method in terms of the simple two-class, one-feature example in Section II-C. As will be recalled from (1), the MAP estimate of a single mean feature value for class 1 is a linear combination of the contributions of observations from class 1, the *a priori* mean of class 1, and the observations from class 2.

Correlation-based tuning exploits the correlations of the mean values of the features in each individual decision class with the mean values of the features in a composite "grand class," class $*$, which consists of the union of all the decision classes. By definition, all observations presented to the classifier belong to class $*$. After each utterance is presented to the classifier, we calculate new estimated mean feature values for each of the actual decision classes by imagining for the kth such class that no observation from class k has yet been obtained (i.e., $N_k = 0$) and that N, the total of all observations, have been obtained from class $*$. The calculations are further simplified by ignoring feature-to-feature correlations of the means of the feature values within a given decision class in estimating new mean values for the features. (We do assume that the observed feature values may be correlated within a given decision class in classifying the utterances.)

If we assume two decision classes, class k and class $*$, with $N_k = 0$ and $N_* = N$, application of (1) produces the correlation-based estimate for the mean of class k:

$$\hat{M}_k = M_{0k} + \frac{N\rho_k\zeta_k\zeta_*}{\sigma_*^2 + N\zeta_*^2}[A_* - M_{0*}] \qquad (5)$$

where

- $\hat{M}_k$ correlation-based estimate of the mean feature values for class k,
- M_{0k} initial mean of this feature for class k,
- A_* sample mean of all observations from all decision classes,
- N total number of observations obtained thus far by the classifier,
- ρ_k correlation coefficient between the random variables M_k and M_*,
- σ_*^2 variance of the observation from class $*$,
- ζ_*^2 speaker-to-speaker variance of M_*,
- ζ_k^2 kth diagonal term of the mean cross covariance matrix (i.e., the speaker-to-speaker variance of M_k).

The term $\rho_k \zeta_k \zeta_* = E[(M_k - M_{0k})(M_* - M_{0*})]$ represents the covariance between M_k and M_* and is equal to the average of the terms in the kth row of the mean cross covariance matrix. Similarly, ζ_*^2 is equal to the average of all the terms in the mean cross covariance matrix.

Before any utterances are observed, $\hat{M}_k$ is set to M_{0k}. We note that the dependences of $\hat{M}_k$ on ρ_k, ζ_k, ζ_*, and σ_* are all reasonable. Specifically, the amount by which the current estimate of $\hat{M}_k$ varies with the sample mean of the composite "grand class" is proportional to ρ_k, the correlation coefficient between the mean values of the given feature and the grand class. For large N, adjustment of $\hat{M}_k$ is also proportional to ζ_k / ζ_* which is the ratio of the speaker-to-speaker variance of the means of M_k to the variance of the means of the grand class. For intermediate values of N, this rate of adjustment is also limited by the parameter σ_*^2, which reflects the variability of the observations.

Applying (2) to classes k and $*$, we obtain the mean square error ζ_k^2 of the estimate $\hat{M}_k$:

$$E\left[(\hat{M}_k - M_k)^2\right] \equiv \zeta_{Nk}^2 = \zeta_k^2 - N \frac{\rho_k^2 \zeta_k^2 \zeta_*^2}{\sigma_*^2 + N\zeta_*^2}. \quad (6)$$

If we repeat this process for each of the C classes, we obtain estimates of the mean value as well as the mean square error for these estimates. These values are then used in performing subsequent classifications. Specifically, if y_k is an observation value from class k, and we assume $p(y_k | M_k)$ to be a normal density function of the form $\mathfrak{N}(M_k, \sigma_k^2)$, we then assume $p(M_k | x_1, x_2, \cdots, x_n)$ to be a normal density function of the form $\mathfrak{N}(\hat{M}_k, \zeta_{Nk}^2)$. Therefore, $p(y_k | x_1, x_2, \cdots, x_n)$ is assumed to be of the form $\mathfrak{N}(\hat{M}_k, \sigma_k^2 + \zeta_{Nk}^2)$. This adaptation process implicitly assumes that the various decision classes in the test set are present with the same frequency as they are in the training set, which is easily accomplished with a vocabulary that consists of the English letters. The correlation-based tuning procedure can also (in principle) be applied to tasks in which the decision classes are not equiprobable, as long as the various decision classes are present with the same relative frequency in the test data as they are in the training data.

By iterating this procedure for each of the D features, we obtain estimates of the class-conditional mean vectors and covariance matrices, and thus the new class-conditional pdf's to be used in subsequent classifications.

C. Sample Results—Acoustically Confusable Letters

We first compared the performance of two methods of tuning without feedback in classifying two sets of acoustically confusable letters of the English alphabet: the E set (**B, C, D, E, G, P, T, V,** and **Z**) and the letters **M** and **N**. In each of these analyses, we presented to the classifier only letters from one of these two confusable sets, and all possible letters are classified at a single node of the decision tree. Classifications were made on the basis of 13 features for the E set and 7 features for the letters **M** and **N**. The data for these evaluations consisted of 4 utterances of each letter to be classified by each of 42 speakers. These data were divided into 3 sets of 14 speakers, and the system was trained on 2 sets and tested on the third. The results presented here represent the performance averaged over the 3 sets of test speakers. As in the evaluations of Section III-D, we averaged over forward and reverse presentation orders of the 4 utterances of each letter from each speaker, so that each iteration in Fig. 6 represents 756 classifications [(9 letters) * (3 sets) * (2 presentation orders) * (14 speakers)] for a 95 percent confidence interval of about ±2 percent, and each iteration in Fig. 7 represents 168 classifications (producing 95 percent confidence intervals of about ±4.5 percent).

Fig. 6 compares the error rate of E-set classifications using no tuning to the speaker, confidence-based tuning, correlation-based tuning, and tuning with feedback, all as a function of how many times the sets of letters to be classified have been presented to the classifier. The confidence-based tuning procedure used to obtain the data in Fig. 6 had a probability threshold of 0.00, which means that the system tuned after every utterance and assumed that it had always made a correct decision. In all of the tuning procedures, the system made use of class-to-class but not feature-to-feature correlations of the means. This was necessitated by the computational problems discussed in Section II-E. With the exception of the first iteration of correlation-based tuning in Fig. 6, both tuning procedures without feedback produce a level of classification performance that falls between the performance achieved with no tuning and the performance achieved with tuning using letter-by-letter feedback from the speaker. The confidence-based tuning procedure is somewhat more successful than the correlation-based procedure in reducing error rate.

Fig. 7 shows similar comparisons, but for classification error rates for the letters **M** and **N**. In this case, the correlation-based tuning procedure performs much better, and in fact equals the performance achieved by tuning with user feedback after four presentations of the letters. Confidence-based tuning with a threshold of 0.00 produces an

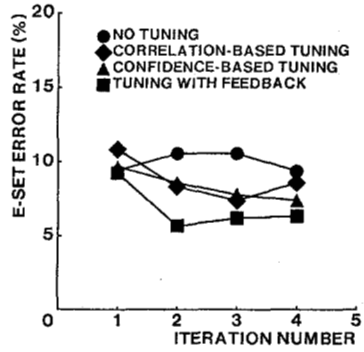


Fig. 6. Comparison of E-set error rates using four different tuning procedures. The confidence-based procedure used a probability threshold of 0.00.

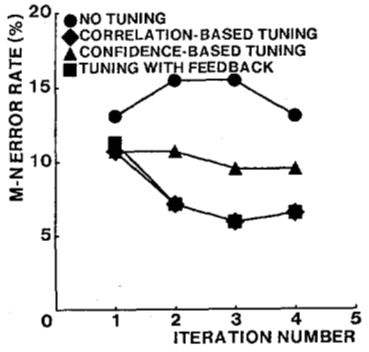


Fig. 7. Comparison of error rates for **M** and **N** using four different tuning procedures.

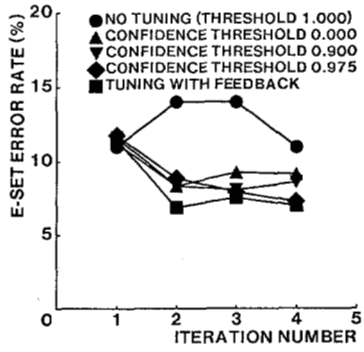


Fig. 8. Dependence of E-set error rates on the threshold parameter in confidence-based tuning.

error rate halfway between that obtained with correlation-based tuning and that obtained with no tuning.

In Fig. 8, we examine the dependence of E-set error rate on the values of the threshold parameter used in the confidence-based procedure, with a smaller number of features than were used to obtain the data in Fig. 6. As the threshold *a posteriori* probability is increased from 0.000 to 0.975, the error rate after four presentations monotonically decreases, and equals the performance obtained tuning with feedback when the threshold is set to 0.975.

D. Discussion—Tuning Without Feedback

Using correlation-based tuning on the letters **M** and **N** and confidence-based tuning (with a threshold of 0.975) on the E set, we have observed classification performance without user feedback that essentially equals the performance obtained on the same data sets with trial-by-trial feedback from the user. With all no-feedback tuning procedures considered produced a lower error rate than that obtained with no tuning, the relative performance of the procedures clearly depends on which set of letters is being considered.

We believe that the choice of an appropriate tuning procedure can be guided by statistical attributes of the letters to be classified. Specifically, we expect that the general performance of the confidence-based tuning procedures will depend on the untuned classification error rate for the letters since the system is likely to tune incorrectly after it makes a classification error. We believe that confidence-based tuning is more successful for the E set than it is for the letters **M** and **N** because the untuned error rate is lower. Similarly, we expect that the performance of the correlation-based tuning procedure will be better when the mean values of the features for the various letters from a given speaker are highly correlated. An examination of the terms of the mean cross covariance matrices confirms that the mean feature values exhibit somewhat greater letter-to-letter correlation for the letters **M** and **N** than for the E-set letters. We conjecture that an index of the relative performance to be expected from correlation-based versus confidence-based tuning procedures for a particular set of letters could be constructed from the ratio of untuned error rate to letter-to-letter correlation of the mean feature values. While this index holds true for the letters considered in the present study, we have not yet explored its validity in more general cases.

It can be seen from Fig. 8 that classification performance using the confidence-based tuning procedure is a nonmonotonic function of the confidence threshold (since performance obtained with a threshold setting of 0.975 is better than that obtained with either 0.0 or 1.0). The confidence threshold providing best performance (0.975 in this case) appears to depend on the set of letters to be classified, and we have not yet been able to predict the optimum threshold from the statistics of these letters *a priori*.

E. Sample Results—Complete Alphabet

We also performed some preliminary experiments in which we extended the procedures for dynamic speaker adaptation without user feedback to the classification of letters from the entire English alphabet. This is a more difficult problem than that of classifying acoustically confusable subsets of letters since, for example, features designed to separate the letters **M** and **N** do not respond in a predictable fashion to letters of the E set. Hence, any tuning procedure without user feedback should be in-

TABLE II
TUNING PROCEDURES AND CONFIDENCE THRESHOLDS CHOSEN FOR EACH NODE IN TUNING THE ENTIRE ALPHABET WITHOUT FEEDBACK

Node Number	Tuning Method	Threshold
1	Confidence	0.975
2	None	
3	Correlation	
4	Confidence	0.900
5	Confidence	0.975
6	Confidence	0.975
7	Correlation	
8	Correlation	
9	Confidence	0.800
10	Confidence	0.900
11	Correlation	
12	None	

voked only when a letter is recognized with extremely high confidence at all nodes of the decision tree. Tuning could then be implemented only for features that are relevant to the classification of that particular letter, with the choice between the confidence-based versus correlation-based procedure varying from feature to feature.

In the present evaluations, we adapted the system's classification of the entire alphabet by applying either the confidence-based or the correlation-based adaptation procedure at the individual nodes of the decision tree shown in Fig. 1. The procedure used and the specific threshold probabilities used at the nodes employing the confidence-based procedure were selected empirically to minimize classification error. These choices and parameter values are summarized in Table II. Finally, tuning was applied only when the *a posteriori* probabilities observed at all nodes of the decision tree were above a fixed threshold of 0.7. We note that the confidence-based procedure proved to be more successful for most of the E-set letters and that the correlation-based procedure was better for the letters **M** and **N**, as might be expected from the results of Section IV-C.

Classification performance was obtained for the same 42 speakers used in the evaluations described in Section IV-C, using the same training and testing procedures as before. Unfortunately, these same data had also been used previously to determine the best tuning procedure for each node and to set the threshold parameters for the nodes with confidence-based tuning. The resulting error rates (for tuning without feedback) are probably somewhat lower than those that would have been obtained using completely new data, but no other corpus of data was available for this study.

Fig. 9 compares recognition performance obtained for the entire alphabet using no tuning, tuning with user feedback (using the **type C** tuning procedure described in Section III), and tuning without user feedback (using the combination of the confidence-based and correlation-based procedures described in this section). Each data

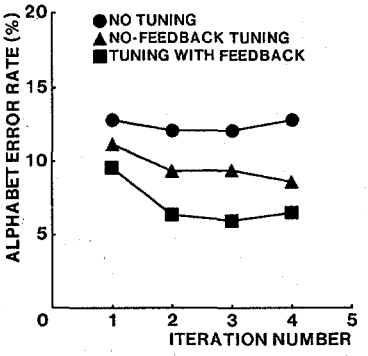


Fig. 9. Comparison of error rates for the entire alphabet using no tuning, tuning without feedback, and tuning with feedback.

TABLE III
NODE-BY-NODE ADAPTATION STATISTICS OBTAINED WHILE CLASSIFYING THE ENTIRE ALPHABET WITHOUT FEEDBACK FROM THE USER. "PERCENTAGE CORRECT" REFERS TO THE PERCENTAGE OF TIMES THE UTTERANCE HAD BEEN CLASSIFIED CORRECTLY WHEN A GIVEN NODE IS TUNED. PERCENTAGES FOR NODES USING THE CORRELATION-BASED METHOD ARE ENCLOSED IN PARENTHESES

Node Number	Calls	Correct Tunings	Incorrect Tunings	Percentage Incorrect
1	8736	6388	414	6.09
2	6715	0	0	N/A
3	1186	944	36	(3.67)
4	3657	2794	205	6.84
5	3068	1901	137	6.72
6	2375	1617	121	6.96
7	706	430	65	(13.13)
8	1036	609	60	(8.97)
9	666	510	6	1.16
10	1724	1061	82	7.17
11	635	538	34	(5.94)
12	667	0	0	N/A

point in Fig. 9 represents 2184 classifications [(26 letters) * (3 sets) * (2 presentation orders) * (14 speakers)], producing a 95 percent confidence interval of about ± 1.25 percent. The use of tuning without user feedback resulted in a 31 percent decrease (to 8.6 percent) in the observed error rate after four presentations of the alphabet, compared to the decrease of 49 percent (to 6.5 percent) that was obtained using tuning with user feedback. To the extent that these results are generally valid, the error rate using tuning without feedback may truly be regarded as *speaker independent* since the system was not provided with any explicit training data for individual users.

Some insight into what factors limit the performance of the adaptation without user feedback may be obtained by the statistics of Table III, which tabulates the number of times each node of the decision tree was reached by the classifier, the number of times each node was tuned when the correct letter was hypothesized by the system, the number of times the nodes were tuned when an incorrect letter was hypothesized, and the percentage of tunings at each node that were based on an incorrect classification.

This particular tabulation tends to overstate the rate of "incorrect tunings." For example, if a **B** is classified as a **D** because of an error at node 10, nodes 2, 4, and 6 would still be tuned correctly, but these nodes would be tabulated as "incorrect tunings" in Table III. The rate of "incorrect tunings" is irrelevant for the correlation-based procedure, and these percentages are enclosed in parentheses. We note from Table III that when the confidence-based method is used, the individual nodes of the decision tree are tuned incorrectly no more than about 7 percent of the time. Furthermore, about 33 percent of the time some important nodes are not tuned at all because classifications are frequently made with insufficiently high confidence. From these observations, we believe that classification performance could continue to improve with further adaptation if more utterances of each letter were presented to the system.

V. General Summary and Conclusions

We have studied the performance of several learning techniques that enable the CMU isolated letter recognition system FEATURE to dynamically adapt to the acoustical characteristics of new speakers. We have assumed that the most significant speaker-to-speaker variability is in the mean values of the features, which are estimated for individual speakers using Bayesian techniques. An important and novel aspect of our work is that the estimation procedure makes use of letter-to-letter correlations of the mean values, as well as the mean values and the covariances of the means, and of the observations in obtaining new estimates.

In evaluations in which the system was informed which letter had actually been uttered, we found that the use of the adaptation procedure reduced the speaker-averaged error rate for the isolated English letters by a factor of 2 after only four presentations of the alphabet. In further experiments with acoustically confusable subsets of the alphabet, we found that the incorporation of letter-to-letter correlations in the adaptation algorithm appeared to be especially helpful during the first two presentations of the letters to be classified. After three or four presentations of the letters, the classification performance obtained using only the sample means of the observed feature values was almost as good as the performance obtained with classification procedures that made use of *a priori* information about these mean values as well.

We also introduced two procedures by which the mean feature values could be learned in an unsupervised fashion. The first of these procedures assumed that decisions made with sufficiently high *a posteriori* probability were correct decisions, and this procedure could produce error rates for the letters of the E set that were as low as the error rates obtained with supervised adaptation. The second procedure adjusted the mean values of individual features according to their correlation with the mean value of all the features, and this procedure produced an error rate for the letters **M** and **N** that was as low as the error rate obtained with supervised adaptation. We also applied these procedures to classification of the entire alphabet and obtained a 31 percent reduction of the error rate, to 8.6 percent after four presentations of the alphabet. We believe that further reductions in error rate may occur after further speech samples are input by a given speaker.

We believe that these MAP learning procedures provide a method by which a significant amount of speaker adaptation can be achieved relatively rapidly, even for a system that is nominally designed to be speaker independent. To our knowledge, this study is the first attempt to include statistical correlations across decision classes in a speaker adaptation procedure explicitly. We are particularly encouraged that the general statistical approach appears to be extensible to unsupervised learning environments because dynamic speaker adaptation for continuous speech recognition systems normally have to operate without user feedback. We are currently investigating several ways of extending these procedures to continuous speech.

Acknowledgment

The authors thank R. A. Cole for providing the initial motivation for this research, specifically including correlated decision classes in the formulation of the problem, as well as for general help and advice on feature-based speech recognition. They also thank D. R. Reddy for a stimulating working environment and computing resources.

References

[1] B. T. Lowerre, "Dynamic speaker adaptation in the Harpy speech recognition system," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1977, pp. 788–790.

[2] R. Zelinski and F. Class, "A learning procedure for speaker-dependent word recognition systems based on sequential processing of input tokens," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1983, pp. 1053–1056.

[3] Y. Grenier, "Speaker adaptation through canonical correlation analysis," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1980, pp. 888–892.

[4] Y. Grenier, L. Miclet, J. C. Maurin, and H. Michel, "Speaker adaptation for phoneme recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1981, pp. 1273–1275.

[5] P. F. Brown, C.-H. Lee, and J. C. Spohr, "Bayesian adaptation in speech recognition," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1983, pp. 761–764.

[6] R. A. Cole, R. M. Stern, M. S. Phillips, S. M. Brill, A. P. Pilant, and P. Specker, "Feature-based speaker-independent recognition of isolated English letters," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1983, pp. 731–734.

[7] R. A. Cole, R. M. Stern, and M. J. Lasry, "Performing fine phonetic distinctions: Templates vs. features," in *Invariance and Variability of Features in Spoken English Letters*, J. Perkell *et al.*, Eds. New York: Lawrence Erlbaum, 1986.

[8] R. L. Kirlin, "*A posteriori* estimation of vocal tract length," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-26, pp. 571–574, 1978.

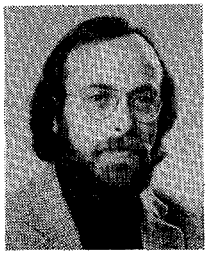
[9] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[10] M. J. Lasry and R. M. Stern, "*A posteriori* estimation of correlated jointly Gaussian mean vectors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 530–535, 1984.

[11] M. J. Lasry, "Dynamic adaptation of statistical parameters in a feature-based isolated letter recognition system," Master's thesis, Dep. Elec. Eng., Carnegie-Mellon Univ., Pittsburgh, PA, 1982.

[12] L. Kanal, "Patterns in pattern recognition: 1968–1974," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 697–722, 1974.

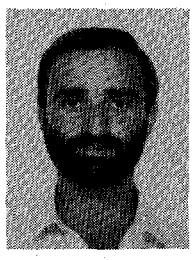
[13] D. G. Keehn, "A note on learning for Gaussian properties," *IEEE Trans. Inform. Theory*, vol. IT-11, pp. 126–132, 1965.



Richard M. Stern (M'77) was born in New York, NY, on July 5, 1948. He received the S.B. degree from the Massachusetts Institute of Technology, Cambridge, in 1970, the M.S. degree from the University of California, Berkeley, in 1972, and the Ph.D. degree from the Massachusetts Institute of Technology in 1977, all in electrical engineering.

He has been on the Faculty of Carnegie-Mellon University (CMU) since 1977, where he is currently an Associate Professor of Electrical Engineering. He was one of the developers of CMU's feature-based system that recognized isolated English letters on a speaker-independent basis, specializing in statistical classification procedures and dynamic speaker adaptation. He is currently working on problems in sentence parsing, speaker adaptation, and speech enhancement for CMU's new generation continuous speech recognition system. He also conducts research in psychoacoustics, concentrating on aspects of binaural perception, complex pitch, and musical timbre.

Dr. Stern is a member of the Acoustical Society of America and the Audio Engineering Society.



Moshé J. Lasry (M'84) graduated from the École Polytechnique and the École Nationale Supérieure des Télécommunications, Paris, France, in 1979 and 1981, respectively, and in 1982 received the M.Sc. degree in electrical engineering from Carnegie-Mellon University (CMU), Pittsburgh, PA.

In 1983 he worked with the Speech Recognition Group of CMU, working on speaker adaptation and speaker-independent isolated word recognition. In 1984 he joined the Communications Division at Tadiran, Holon, Israel, where he worked on low-bit rate speech coding. He is presently in charge of research and development in speech recognition at Calltalk Ltd., Tel-Aviv, Israel.