DIGITAL SIGNAL PROCESSING 2, 157–173 (1992)

Dimensionality Reduction of the Enhanced Feature Set for the HMM-Based Speech Recognizer

K. K. Paliwal*

Speech Research Department, AT&T Bell Laboratories, Murray Hill, New Jersey 07974

1. INTRODUCTION

In the past few years, a great deal of research has been directed toward finding acoustic features that are effective for automatic speech recognition. Until recently, most of the speech recognizers used about 12 cepstral coefficients derived through the linear prediction analysis as recognition features [1]. In [2,3], Furui investigated the use of temporal derivatives of cepstral coefficients and energy as recognition features in a dynamic time warping-based isolated word recognizer and showed how the recognition performance improves with the inclusion of first derivatives in the feature set. These results were later confirmed in a number of studies for more general tasks (such as speaker-independent connected digit recognition and large-vocabulary continuous speech recognition) using the hidden Markov model (HMM)-based speech recognizers [4–6]. More recently, some studies which advocate the use of second (and higher)-order temporal derivatives of cepstral coefficients for speech recognition have been reported [7–9]. These temporal derivatives have also been found useful as recognition features for speaker recognition [10–12]. As a result, most of the present-day speech recognizers use a larger feature set for enhancing the speech recognition performance [13–15]. This feature set usually consists of cepstral coefficients and energy, and their derivatives.

Though the addition of new features has improved the speech recognition performance, it has created some problems, too. For example, the recognizer using a larger (or, enhanced) feature set is computationally more complex and requires more storage which makes its real-time implementation more difficult and costly. In addition, a large amount of data is needed for training the recognizer if the size of the feature set is increased [16]. In order to avoid these problems, we try in this paper to reduce the dimensionality of the enhanced feature set without affecting the recognition performance. For this, we study a number of dimensionality reduction methods. The speech recognizer employed in this study uses continuous density HMMs with Gaussian mixture densities, where the covariance matrices are assumed to be diagonal. The recognizer is applied here to a multispeaker isolated word recognition task. The problem of dimensionality reduction is studied here specifically for the enhanced feature set used with the speech recognition systems at the AT&T Bell Laboratories [13,14]. This feature set consists of the following 38 features: 12 cepstral coefficients and 12 first derivatives and 12 second derivatives of these coefficients (known as the delta cepstrum and delta-delta cepstrum coefficients, respectively), and the first derivative and the second derivative of energy (known as the delta energy and delta-delta energy, respectively). Thus, we have here a D-dimensional feature space where $D = 38$, and our aim is to reduce the dimensionality of this feature space to d ($<D$) without sacrificing in terms of recognition performance.

There are a number of methods reported in the pattern recognition literature for reducing the dimensionality of a feature space [17]. Though some of these methods have been studied exhaustively in the past for the speaker recognition application [18–24], they have been applied to speech recognition only very recently by a few researchers [25–29]. In the present paper, we study four different dimensionality reduction methods for speech recognition. The first two methods select the features from the original set by

* Current address: Computer Systems and Communications Group, Tata Institute of Fundamental Research, Homi Bhabha Road, Bombay 400 005, India.

1051-2004/92 $4.00

using the F-ratio (i.e., the ratio of between-class and within-class variances) and the recognition rate on training data for rank-ordering the individual features. The last two methods are based on the linear discriminant analysis and the principal component analysis. We show in this paper that it is possible to reduce the dimensionality of the 38-dimensional feature space to 20 without losing in terms of recognition performance.

The paper is organized as follows. In Section 2, different dimensionality reduction methods used in this paper are described. The speech recognition system is described in Section 3. This includes the computation of recognition features from the speech signal, characterization of HMMs, estimation of HMM parameters from the training data, and recognition of the input speech utterance using this recognizer. Recognition experiments are conducted to evaluate different dimensionality reduction methods. These recognition experiments and their results are described in Section 4. In Section 5, these results are discussed. In Section 6, conclusions are reported.

2. DIMENSIONALITY REDUCTION METHODS

A number of methods are reported in the pattern recognition literature for reducing the dimensionality of a feature space [17]. These methods can be grouped into two categories: feature selection methods and feature extraction methods. The feature selection methods (also known as "subsetting" methods [21]) reduce dimensionality by selecting a subset of the original feature set. The feature extraction methods (also known as the "transformation" methods) are more general in the sense that they reduce the dimensionality by projecting the original D-dimensional feature space on a d-dimensional subspace (where $d < D$) through a transformation. Note that each feature in the reduced feature set is a part of the original feature set when the feature selection methods are used for dimensionality reduction. On the contrary, when the feature extraction methods are used for dimensionality reduction, each feature in the reduced feature set is a combination of all the features in the original feature set. Thus, the feature selection methods reduce the computational complexity by not computing those features which are not in the reduced feature set, while this is not possible with the feature extraction methods where all the D features have to be computed before the dimensionality reduction is performed through transformation.

In the feature selection methods, selection of features is done by devising a figure of merit which reflects the goodness of an individual feature in the recognition task. Once the figure of merit is chosen, it is used for rank-ordering D features of the original set. Top d ($<D$) features are selected from the rank-ordered list. Note that this procedure finds only the subset of d individually best features. This subset is, in general, not the same as the best subset of d features. In order to find the best subset, one has to inspect all possible subsets, which is computationally very expensive and has not been done in this paper.

Efficiency of a feature selection method depends on how good is the figure of merit in reflecting the effectiveness of individual features. A number of figures of merit have been reported in the literature for feature selection purposes [17]. Some of these figures of merit are F-ratio (ratio of between-class and within-class variances), Mahalanobis distance, Bhattacharya distance, Matusita distance, Patrick–Fisher distance, divergence measure, mutual information measure, and entropy measure. We use in this paper two feature selection methods. The first method uses the F-ratio as the figure of merit for feature selection. Since the goal of a pattern recognizer is to recognize the input patterns correctly, the relative merit of a feature should be judged by its contribution to recognition performance. Therefore, we use in the second method the recognition rate on training data as the figure of merit for feature selection.

As mentioned earlier, the feature extraction methods reduce dimensionality by projecting the original feature space on a smaller subspace through a transformation. Though this transformation can be linear or nonlinear, we use in this paper only linear transformations. A number of transformations are reported in the literature for feature extraction purposes [17]. We use here two linear transformations derived through the linear discriminant and the principal component analyses for feature extraction.

Thus, in this paper, we study four methods for reducing the dimensionality of the feature space. The first two of them are the feature selection methods using the F-ratio and the recognition rate on training data as figures of merit. The last two are the feature extraction methods using the linear discriminant and the principal component analyses for transforming the input pattern from the original feature space to a lower dimensional space. These methods are described below.

2.1. Method 1: Feature Selection Using the F-Ratio as the Figure of Merit

The F-ratio has been widely used as the figure of merit for feature selection in speaker recognition applications [18–22]. It is defined as the ratio of the between-class variance and the within-class variance. In statistical literature [30], this ratio is commonly

used as a statistic in the analysis of variance for determining whether or not significant differences exist among the means of several groups of observations, where each group follows a Gaussian distribution. In the context of feature selection for pattern classification, it tries to select the feature which maximizes the separation between different classes and minimizes the scatter within the classes.

When the F-ratio is used as figure of merit for dimensionality reduction, the following assumptions have to be satisfied: (1) The feature vectors (or, patterns) within each class must have Gaussian distribution, (2) the features should be statistically uncorrelated, and (3) the variances within each class must be equal. Since the variances within each class are generally not equal, the pooled within-class variance is used to define the F-ratio. If the number of training patterns in each of the K classes is assumed to be the same ($=N$),[1] the F-ratio for the ith feature is defined as

$$F_i = \frac{B_i}{W_i}, \tag{1}$$

where B_i is the between-class variance and W_i the pooled within-class variance. These variances are given by

$$B_i = \frac{1}{K}\sum_{k=1}^{K}(\mu_{ik} - \mu_i)^2, \tag{2}$$

$$W_i = \frac{1}{K}\sum_{k=1}^{K} W_{ik}, \tag{3}$$

where μ_{ik} and W_{ik} are, respectively, the mean and variance of the ith feature for the kth class, and μ_i is the overall mean of the ith feature. These are given by

$$\mu_{ik} = \frac{1}{N}\sum_{n=1}^{N} x_{ikn}, \tag{4}$$

$$W_{ik} = \frac{1}{N}\sum_{n=1}^{N}(x_{ikn} - \mu_{ik})^2, \tag{5}$$

$$\mu_i = \frac{1}{K}\sum_{k=1}^{K}\mu_{ik}, \tag{6}$$

where x_{ikn} is the ith feature of the nth training pattern from the kth class.

[1] Note that this assumption is made here for simplifying the presentation. It can be easily extended to an unequal number of training patterns in different classes.

2.2. *Method 2: Feature Selection Using Recognition Rate on Training Data as the Figure of Merit*

As mentioned earlier, the goal of a pattern recognizer is to recognize the unknown pattern correctly. This means that the relative merit of a feature should be judged by its contribution to the recognition performance. In order to accomplish it, we use the recognition rate on the training data as the figure of merit. Here, each feature is used individually to recognize all the patterns in the training set using the pattern recognizer under study and the recognition rate is computed for each of the D features in the original space. The D features are rank-ordered using this recognition rate as the figure of merit, and top d ($<D$) features are selected.

It is clear that the d features selected by this method depend on the pattern recognizer used for computing the recognition rate on the training data. Therefore, these d features will be relevant only for this pattern recognizer. However, this method has the advantage that it makes fewer assumptions than the F-ratio-based method. It makes the assumption that the features are statistically uncorrelated, but the other assumptions, that the within-class distributions are Gaussian and the within-class variances are equal, are not needed.

2.3. *Method 3: Feature Extraction Using Linear Discriminant Analysis*

In this method, dimensionality reduction is accomplished by projecting the original D-dimensional space on a d-dimensional subspace (where $d < D$) and finding a linear transformation that defines this subspace using the linear discriminant analysis. The linear transformation is computed here in such a way that it maximizes the F-ratio of the training data in the transformed subspace. Thus, linear discriminant analysis can be considered to be a generalization of the F-ratio-based method. Here, the assumption regarding the independence of features is not needed. However, two other assumptions are still made; namely, the within-class distributions are Gaussian, and the within-class covariance matrices are equal. Linear discriminant analysis has been applied in the past to reduce the dimensionality of a feature set for speaker and speech recognition applications [20,21,23,25–27].

In linear discriminant analysis, the linear transformation is defined in terms of a rank-ordered set of linearly independent vectors, $\mathbf{u}_i$, $i = 1, \ldots, d$. The first of these vectors, $\mathbf{u}_1$, is the direction in the original D-dimensional feature space which, when the training patterns are projected onto it, produces the

maximum value of the F-ratio (i.e., the ratio of between-class and within-class variances). The second vector $\mathbf{u}_2$ is chosen such that it is linearly independent to $\mathbf{u}_1$ and produces the next largest F-ratio. This process is repeated until all the d linearly independent vectors, $\mathbf{u}_i$, $i = 1, \ldots, d$, are found. It can be shown [30] that these vectors can be computed as the d eigenvectors corresponding to d largest eigenvalues of the matrix $\mathbf{W}^{-1}\mathbf{B}$, where $\mathbf{B}$ is the between-class covariance matrix and $\mathbf{W}$ the pooled within-class covariance matrix. These matrices are symmetric and can be computed from the training data as

$$\mathbf{B} = \frac{1}{K}\sum_{k=1}^{K}(\mu_k - \mu)(\mu_k - \mu)^t, \tag{7}$$

$$\mathbf{W} = \frac{1}{K}\sum_{k=1}^{K}\mathbf{W}_k, \tag{8}$$

where μ_k and $\mathbf{W}_k$ are the mean vector and covariance matrix of the kth class, respectively, and μ is the overall mean. These are given by

$$\mu_k = \frac{1}{N}\sum_{n=1}^{N}\mathbf{x}_{kn}, \tag{9}$$

$$\mathbf{W}_k = \frac{1}{N}\sum_{n=1}^{N}(\mathbf{x}_{kn} - \mu_k)(\mathbf{x}_{kn} - \mu_k)^t, \tag{10}$$

$$\mu = \frac{1}{K}\sum_{k=1}^{K}\mu_k, \tag{11}$$

where $\mathbf{x}_{kn}$ is the nth training pattern from the kth class.[2]

Thus, in linear discriminant analysis, the linear transformation is given by the matrix $\mathbf{U}^t$, where $\mathbf{U}$ is a $D \times d$ matrix whose columns are the eigenvectors corresponding to the d largest eigenvalues of the matrix $\mathbf{W}^{-1}\mathbf{B}$. It can be shown [23] that the matrix $\mathbf{U}$ is given by

$$\mathbf{U} = \mathbf{C}\mathbf{L}^{-1/2}\mathbf{V}, \tag{12}$$

where $\mathbf{C}$ is an unitary matrix diagonalizing the within-class matrix $\mathbf{W}$ to a diagonal matrix $\mathbf{L}$, i.e., $\mathbf{C}^t\mathbf{W}\mathbf{C} = \mathbf{L}$, and $\mathbf{V}$ is an unitary matrix whose columns are chosen to be the eigenvectors corresponding to the d largest eigenvalues of the symmetric matrix $\mathbf{S} = \mathbf{L}^{-1/2}\mathbf{C}^t\mathbf{B}\mathbf{C}\mathbf{L}^{-1/2}$. From Eq. (12), it is clear that when $d = D$, i.e., no dimensionality reduction is performed, the linear transformation computed through linear discriminant analysis is equivalent to a rotation, followed by scaling and then followed by another rotation. Using this property, it is easy to show that the Mahalanobis distance [17] remains invariant under this transformation, but the Euclidean distance does not remain invariant under this transformation.

[2] Note that the number of training patterns in each of the K classes is assumed to be the same ($=N$) for simplifying the presentation. It can be easily extended to an unequal number of training patterns in different classes.

2.4. *Method 4: Feature Extraction Using Principal Component Analysis*

In this method, dimensionality reduction is achieved by projecting the original D-dimensional feature space on a d-dimensional subspace and finding the orientation of the subspace which best preserves the information available in the original space using the principal component analysis. This method has been used in the past to reduce the dimensionality of the feature space for speech and speaker recognition applications [23,29]. Here, the input pattern in the original D-dimensional feature space is transformed to the Karhunen–Loeve (KL) coordinate system and dimensionality is reduced by representing the pattern by d coordinates in the KL coordinate system. The KL coordinate system represents optimally (in a minimum mean square error sense) a set of D-dimensional patterns (or, vectors) by another set of vectors of a lower dimensionality.

In order to derive the KL transformation, consider all the D-dimensional patterns (or, vectors), $\mathbf{x}_{kn}$, $k = 1, \ldots, K$, $n = 1, \ldots, N$, in the training set. The aim in principal component analysis is to approximate each D-dimensional vector $\mathbf{x}_{kn}$ by a d-dimensional vector $\mathbf{y}_{kn}$ (where $d < D$) such that the mean square error,

$$E = \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}(\mathbf{x}_{kn} - \mathbf{U}\mathbf{y}_{kn})^t(\mathbf{x}_{kn} - \mathbf{U}\mathbf{y}_{kn}), \tag{13}$$

is minimized. Here, $\mathbf{U}$ is a $D \times d$ matrix. It can be shown [30] that E is minimum if the columns of matrix $\mathbf{U}$ are chosen to be the eigenvectors corresponding to the d largest eigenvalues of the total covariance matrix $\mathbf{T}$. This matrix is defined as

$$\mathbf{T} = \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}(\mathbf{x}_{kn} - \mu)(\mathbf{x}_{kn} - \mu)^t, \tag{14}$$

where μ is the overall mean given by

$$\mu = \frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N}\mathbf{x}_{kn}. \tag{15}$$

Thus, in principal component analysis, the linear transformation is given by the matrix $\mathbf{U}^t$, where $\mathbf{U}$ is a $D \times d$ unitary matrix whose columns are the eigen-

vectors corresponding to the d largest eigenvalues of the total covariance matrix **T**. When no dimensionality reduction is performed (i.e., $d = D$), this transformation amounts to a rotation in the feature space. This means that in this case the Mahalanobis distance and the Euclidean distance remain invariant under this transformation.

Note that, in this method, the information available in the training data about the class labels of the training patterns is totally ignored in the computation of the total covariance matrix **T** (see Eq. (14)). Also, this method neither maximizes the between-class separation nor minimizes the within-class scatter. It only minimizes the mean square error in approximating the set of training vectors by another set of vectors of a lower dimensionality. However, it can be shown [17] that the total covariance matrix is a sum of the between-class and the within-class covariance matrices. Usually the total covariance matrix is dominated by the between-class covariance matrix; i.e., the variance in the training data is due mainly to separation between the classes. In such cases, the KL transform preserves most of the class separation. But, it is not always guaranteed.

3. THE HMM-BASED ISOLATED WORD RECOGNIZER

In this section, we provide a brief overview of the basic recognition system which is used for evaluating the dimensionality reduction methods. This system uses continuous density HMMs with Gaussian mixture densities, where the covariance matrices are assumed to be diagonal. The system uses the following 38 recognition features: 12 cepstral coefficients, 12 delta cepstral coefficients, 12 delta-delta cepstral coefficients, 1 delta energy, and 1 delta-delta energy. These features are used in the current speech recognition systems at the AT&T Bell Laboratories [13,14]. The key elements of the HMM-based isolated word recognition system are described in the following subsections.

3.1. Computation of Recognition Features

As mentioned earlier, the recognizer uses a set of 38 features for speech recognition. These features include 12 cepstral coefficients, 12 delta cepstral coefficients, 12 delta-delta cepstral coefficients, 1 delta energy, and 1 delta-delta energy. Linear predictive coding (LPC) analysis is used here to compute the cepstral features. The processing steps for computing the 38 recognition features from the speech signal are briefly described below. (For more details, see [4,13].)

1. Preemphasis: The speech signal (digitized at 6.67 kHz sampling rate) is preemphasized using a simple first-order finite impulse response digital filter, whose transfer function is defined by $H(z) = 1 - \alpha z^{-1}$, where α is the preemphasis factor. (We use $\alpha = 0.95$ in our implementation.)

2. Blocking into frames: The preemphasized speech signal is analyzed framewise, where each frame consists of N_w consecutive speech samples. The successive frames are separated by N_s samples. (In our implementation, N_w corresponds to 45 ms of the signal, and N_s to 15 ms.)

3. LPC analysis: A pth-order LPC analysis is performed for each frame. Prior to LPC analysis, N_w samples of the preemphasized speech signal are weighted by a Hamming window function to avoid spectral leakage [31]. The first $p + 1$ autocorrelation coefficients are computed from the windowed speech signal and are used to compute the p LPC coefficients using the Levinson–Durbin recursion method [31]. (We use $p = 8$ in our implementation.)

4. Computation of cepstral coefficients: From the p LPC coefficients, Q cepstral coefficients (where $Q > p$) are computed through a recursion relation [31]. (In our implementation, we use $Q = 12$.) These Q cepstral coefficients are weighted by the cepstral window (or, lifter) [32] as

$$C_t(m) = \left[1 + \frac{Q}{2}\sin\left(\frac{\pi m}{Q}\right)\right] c_t(m), \quad 1 \le m \le Q, \quad (16)$$

where $c_t(m)$ is the mth cepstral coefficient of the tth frame and $C_t(m)$ is its weighted version. These Q weighted cepstral coefficients are used in the feature set for recognition. In the sequel, we will be referring to the weighted cepstral coefficients simply as the cepstral coefficients.

5. Computation of delta cepstral coefficients: The delta cepstral coefficients (or, the first temporal derivatives of the cepstral coefficients) have been computed in the past either through regression [2,4] or as simple differences [5,7]. Here, we use the regression implementation for computing these derivatives. In the regression implementation, a sequence of each of the Q cepstral coefficients (over a finite window of $(2K + 1)$ frames and centered on the current frame) is approximated by a first-order orthogonal polynomial and the derivative is computed as

$$DC_t(m) = G_1 \sum_{k=-K}^{K} kC_{t+k}(m), \quad 1 \le m \le Q, \quad (17)$$

where $DC_t(m)$ is the mth delta cepstral coefficient of the tth frame, and G_1 is the gain term which is chosen

such that variances of $\{C_t(m)\}$ and $\{DC_t(m)\}$ are equal. (We use $K = 2$ and $G_1 = 0.375$ in our implementation.)

6. Computation of delta-delta cepstral coefficients: The delta-delta cepstral coefficients (or, the second-order temporal derivatives of the cepstral coefficients) are computed from the delta cepstral coefficients by simple differencing as

$$DDC_t(m) = G_2[DC_{t+1}(m) - DC_{t-1}(m)], \quad 1 \le m \le Q, \quad (18)$$

where $DDC_t(m)$ is the mth delta-delta cepstral coefficient of the tth frame, and G_2 is the gain term determined from variance considerations [13]. (We use $G_2 = 0.375$ in our implementation.)

7. Computation of delta energy: Energy of each frame has already been computed as the zeroth autocorrelation coefficient in step 3. We convert this energy in decibel scale and compute the delta energy by computing the first derivative through regression implementation as

$$DE_t = G_3 \sum_{k=-2}^{2} kE_{t+k}, \quad (19)$$

where E_t is the energy of the tth frame in decibels and DE_t is the corresponding delta energy. Here G_3 is the gain term determined from variance considerations [13]. (We use $G_3 = 0.0375$ in our implementation.)

8. Computation of delta-delta energy: The delta-delta energy is computed from the delta energy using the differencing method as

$$DDE_t = G_4[DE_{t+1} - DE_{t-1}], \quad (20)$$

where G_4 is the gain term determined from variance considerations [13]. (We use $G_4 = 0.375$ in our implementation.)

Thus, the feature computation procedure outlined above represents each frame of the speech signal by an overall feature vector which is a concatenation of Q cepstral coefficients, Q delta cepstral coefficients, Q delta-delta cepstral coefficients, one delta energy, and one delta-delta energy. This vector has $3Q + 2$ components and the dimensionality of the feature space is $D = 3Q + 2$. (Since in our implementation $Q = 12$, we have a 38-dimensional feature vector representing each frame.)

It might be noted here that the cepstral weights used in Eq. (16) and the gains G_1, G_2, G_3, and G_4 used in Eqs. (17)–(20) do not affect the likelihood calculations used for scoring the HMMs. Their only role is during the segmental k-means training procedure [33], where the k-means clustering algorithm [34] is used with a Euclidean distance measure to define individual mixtures in the mixture density. If the k-means clustering algorithm is replaced by the maximum likelihood clustering algorithm [35–37], these weights and gains are irrelevant even during the training phase and, hence, can be set to arbitrary positive values (such as unity) without affecting the recognition system.

3.2. HMM Characterization

Each word in the vocabulary is characterized by a left-to-right HMM containing S states. The transition probabilities between the states are denoted by $a_{ij}, i, j = 1, \ldots, S$, where $j - i$ can be 0 (for self-transition within a state) or 1 (for transition to the next state). (Note that the skip transitions are not allowed.) The observation probability for each state is characterized by a continuous probability density function specified as a mixture of Gaussian densities. Thus, the probability of observing the feature vector $\mathbf{O}_t$ at the tth frame in state j is given by

$$b_j(\mathbf{O}_t) = \sum_{m=1}^{M} c_{jm}\mathbf{N}(\mathbf{O}_t, \mu_{jm}, \mathbf{W}_{jm}), \quad (21)$$

where $\mathbf{N}(\mathbf{O}_t, \mu_{jm}, \mathbf{W}_{jm})$ represents a multivariate Gaussian probability density function with mean vector μ_{jm} and covariance matrix $\mathbf{W}_{jm}$ for the mth component of the M-component mixture density for the jth state, and c_{jm}, $m = 1, \ldots, M$, are the mixture weights for the jth state. (In our experiments, we assume the D features in the feature set to be uncorrelated and, hence, use diagonal covariance matrices to specify the Gaussian mixture densities. Also, we use $S = 5$ and $M = 5$ in our implementation.)

3.3. HMM Training

For each word in the N-word vocabulary, an HMM is designed; i.e., the HMM parameters (transition probabilities, mixture weights, mean vectors, and covariance matrices) are estimated from a training set of data representing multiple utterances of the vocabulary word. The segmental k-means training procedure [33] is used to estimate these parameters. The processing steps used in the segmental k-means training procedure are outlined below.

1. Initialization: Segment uniformly each of the training utterances (of a given word) into S segments of equal durations, where S is the number of states in an HMM.

2. Clustering: Partition the set of frames belonging to the jth state into M clusters using the k-means clustering algorithm [34]. Here, each cluster represents one of the M mixtures in the mixture density function, defined by Eq. (21). This step is repeated for all the states, i.e., for $j = 1, \ldots, S$.

3. Estimation: Compute the HMM parameters (transition probabilities, mixture weights, mean vectors, and diagonal covariance matrices) using the segmentation and clustering information from the previous steps.

4. Segmentation: Segment each of the training utterances (of the given word) into S segments using the HMM estimated in the previous step.

5. Iteration: Iterate steps 2–4 until convergence, i.e., until the average model likelihood of the given word on its training utterances converges.

This procedure is repeated for all the N words in the vocabulary to design their HMMs.

3.4. Testing

In the testing phase, an input utterance of an unknown word is decoded by the HMMs of the individual words in the vocabulary using the Viterbi algorithm [33], and the likelihood scores are computed. The maximum likelihood decision rule is applied for recognition; i.e., the utterance is recognized as the word whose HMM shows the highest likelihood score.

4. RECOGNITION EXPERIMENTS AND RESULTS

In this section, we describe speech recognition experiments where different dimensionality reduction methods are evaluated on an isolated word recognition task using the HMM-based recognizer described in the preceding section. We start with $D = 38$ features in the enhanced feature set used at the AT&T Bell Laboratories. Using these methods, we reduce the dimensionality of the feature space to d (where $d < D$). This section is organized as follows. The speech data base used in these recognition experiments is described in Subsection 4.1. In Subsection 4.2, recognition experiments with the original feature set are described to provide baseline results. Recognition results with different dimensionality reduction methods are described in Subsection 4.3.

4.1. Speech Data Base

In order to evaluate the dimensionality reduction methods, we use here an alpha-digit vocabulary of $N = 39$ words consisting of the 26 letters of English alphabet (A–Z), 3 command words (stop, error and repeat), and the 10 English digits (0–9). The data base consists of two sets of data, each consisting of one utterance of each of the 39 words by each of 100 different speakers (50 men and 50 women). One set of data is used for training and another for testing. Since the training and the test data sets use the same set of 100 speakers, this data base allows us to evaluate the performance of the isolated word recognizer in a multispeaker mode. The training and testing tokens were recorded over local dialed-up telephone lines, bandpass filtered to 200–3200 Hz, and digitized at a sampling rate of 6.67 kHz. This data base has been used in the past in a number of recognition experiments. (See [38] for details.)

TABLE 1

Baseline Recognition Results

Feature set	No. of features	Recognition accuracy (in %)
C	12	78.97
$C + DC$	24	86.69
$C + DC + DE + DDE$	26	88.77
$C + DC + DDC + DE + DDE$	38	88.87

4.2. Recognition Experiments with the Original Feature Set

Speech recognition experiments with the original feature set (consisting of 38 features) are conducted to provide a baseline performance, which will be used later as a reference for evaluating different dimensionality reduction methods. In order to facilitate this evaluation process, we also report recognition results with some of the subsets of this feature set. Results are listed in Table 1. In this table, the cepstral coefficients are denoted by C, the delta cepstral coefficients by DC, the delta-delta cepstral coefficients by DDC, the delta energy by DE, and the delta-delta energy by DDE. This table shows how the recognition performance improves with the inclusion of more and more features in the feature set. Using all the 38 features in the original feature set, the isolated word recognizer performs with 88.87% recognition accuracy.

4.3. Recognition Experiments with the Reduced Feature Sets

In this section, we describe speech recognition experiments in which the four dimensionality reduction methods described in Section 2 are studied as to their effectiveness in reducing the dimensionality of the original feature space. Note that in all these experiments, the training data set is used for training the HMMs for each word in the vocabulary as well as for computing the figure of merit or transformation re-

TABLE 2

F-Ratios for the Individual Features in the Original Feature Set

Feature	*F*-ratio	Feature	*F*-ratio	Feature	*F*-ratio	Feature	*F*-ratio
C_1	1.76	DC_1	0.37	DDC_1	0.13	DE	1.92
C_2	2.17	DC_2	0.41	DDC_2	0.09	DDE	0.32
C_3	1.53	DC_3	0.36	DDC_3	0.11		
C_4	0.61	DC_4	0.15	DDC_4	0.05		
C_5	0.39	DC_5	0.15	DDC_5	0.04		
C_6	0.25	DC_6	0.10	DDC_6	0.03		
C_7	0.29	DC_7	0.08	DDC_7	0.04		
C_8	0.39	DC_8	0.11	DDC_8	0.04		
C_9	0.35	DC_9	0.12	DDC_9	0.04		
C_{10}	0.58	DC_{10}	0.11	DDC_{10}	0.04		
C_{11}	0.44	DC_{11}	0.12	DDC_{11}	0.04		
C_{12}	0.16	DC_{12}	0.05	DDC_{12}	0.03		

quired in these dimensionality reduction methods. The test data set is used only to compute the recognition performance of the recognizer with different reduced feature sets. The recognition experiments are described below for each of the four dimensionality reduction methods.

4.3.1. Recognition experiments with the dimensionality reduction method 1. Here, we use the feature selection method based on the *F*-ratio as the figure of merit for dimensionality reduction. The value of the *F*-ratio for each of the $D = 38$ features of the original feature set is computed from the training set data as follows. The HMMs designed from the training data (using all the 38 features) are used to segment the training utterances of all the words into S states using the Viterbi algorithm [33]. For computing the within-class[3] variance of a given feature, the variance of this feature is computed for each of the M mixture components of S states and N words. These variances are pooled together to get the within-class variance of each feature. For computing the between-class variance of a given feature, the mean value of this feature is computed for each of the S states and N words. The statewise variance of these mean values representing different words is computed for all the S states. These statewise between-word variances are pooled together to get the between-word (or, between-class) variance of each feature. The *F*-ratio for each of the D features is computed as the ratio of the between-class to within-class variances. Table 2 lists the *F*-ratios for all the $D = 38$ features in the feature set. Rank-ordering of the 38 features in terms of their *F*-ratios is listed in Table 3. Dimensionality reduction is performed by selecting the top d ($<D$) features from this rank-ordered list. Recognition performance of the isolated word recognizer on the test data set as a function of the size d of the reduced feature set is shown in Table 4. It is clear from this table that we can reduce the dimensionality of the original feature space from $D = 38$ to $d = 24$ without affecting the recognition performance.

4.3.2. Recognition experiments with the dimensionality reduction method 2. Here, we use the feature selection method based on the recognition rate on training data as the figure of merit for dimensionality reduction. In order to estimate this figure of merit for a given feature in the original feature set, the training data set is used to design HMMs of the N vocabulary words using this feature; and the recognition performance is computed on the same training data set using this feature. This procedure is repeated for all the D features and the recognition rates on the training data set using individual features are computed. These are listed in Table 5. Rank-ordering of the $D = 38$ features in the original feature set using the recognition on training data as the figure of merit is shown in Table 6. Dimensionality reduction is done by selecting the top d ($<D$) features from this rank-ordered list. Recognition performance of the recognizer on the test data set is shown in Table 7 as a function of the size of the reduced feature set. This table shows that we can use a reduced feature set of size $d = 16$ and still can get better performance than that obtained by using all the 38 features in the original feature set. Thus, this dimensionality reduction method performs better than method 1 used in the preceding subsection. The reason that this happens is rather obvious. This method uses the recognition rate on training data as the figure of merit which reflects the recognition capability of each feature better than the *F*-ratio used in method 1.

4.3.3. Recognition experiments with the dimensionality reduction method 3. As mentioned earlier, the

[3] Note that the individual words in the N-word vocabulary define the N classes. We use here the terms word and class interchangeably.

TABLE 3

Rank-Ordering of the 38 Features in the Original Feature Set Using the F-Ratio as the Figure of Merit

Rank	Feature	Rank	Feature	Rank	Feature	Rank	Feature
1	C_2	13	C_9	25	DDC_3	37	DDC_6
2	DE	14	DDE	26	DC_6	38	DDC_{12}
3	C_1	15	C_7	27	DDC_2		
4	C_3	16	C_6	28	DC_7		
5	C_4	17	C_{12}	29	DC_{12}		
6	C_{10}	18	DC_5	30	DDC_4		
7	C_{11}	19	DC_4	31	DDC_{10}		
8	DC_2	20	DDC_1	32	DDC_9		
9	C_5	21	DC_9	33	DDC_5		
10	C_8	22	DC_{11}	34	DDC_{11}		
11	DC_1	23	DC_8	35	DDC_8		
12	DC_3	24	DC_{10}	36	DDC_7		

feature selection method 1 which uses the F-ratio as the figure of merit assumes the within-class and the between-class covariance matrices to be diagonal. However, in practice, this assumption does not hold well. For example, we show in Fig. 1 the within-class covariance matrix (computed for the 38 features in the original feature set from the training set data), where the off-diagonal elements are not all zero. The linear discriminant analysis-based feature extraction method does not make this assumption about the diagonality of these matrices and, hence, is expected to perform better for dimensionality reduction. This method is studied in this subsection.

In this dimensionality reduction method, the linear discriminant analysis is used to define a linear transformation which projects the original D-dimensional feature space on a d-dimensional subspace (where $d < D$). In order to compute this transformation, we need the within-class covariance matrix (**W**) and the between-class covariance matrix (**B**). These are computed from the training data set in a manner similar to that used in Subsection 4.3.1. These covariance matrices are used to compute the required transformation U using Eq. (12). Reduced feature sets of different sizes are computed using this method and the recognition performance of the recognizer is evaluated on the test data set for each of these reduced feature sets. Results are shown in Table 8. Comparison of this table with Tables 4 and 7 reveals that this method does not perform as well as methods 1 and 2. It looks surprising that despite using the full covariance matrices, this method did not perform better than the F-ratio-based method. The reason for this is as follows. We have used full covariance matrices in this method for reducing the dimensionality, but the HMM-based recognizer in our experiments uses only diagonal covariance matrices. In fact, when the full covariance matrices are used in the HMM-based recognizer, this method has been found to perform better than the F-ratio-based method for dimensionality reduction. Also, note that the recognition performance of the recognizer using all the 38 features in the transformed space is poorer than that obtained by using all the 38 features in the original feature space. Again, this happens due to the use of diagonal covariance matrices in the HMM-based recognizer.

TABLE 4

Recognition Accuracy as a Function of the Size d of the Reduced Feature Set

Size of reduced feature set	Recognition accuracy in (%)
4	78.21
8	83.21
12	87.10
16	87.15
20	88.18
24	89.03
28	89.56
32	89.26
36	89.69
38	88.87

Note. Dimensionality reduction is done by using the F-ratio as the figure of merit.

4.3.4. Recognition experiments with the dimensionality reduction method 4. Here, we study the principal component analysis-based feature extraction method for dimensionality reduction. Principal component analysis is used to compute a linear transformation which projects the original D-dimensional feature space on a d-dimensional subspace (where $d < D$). Here the total covariance matrix **T** is computed from the training data set using Eq. (14) and the linear transformation U that reduces the dimensionality to d is obtained as the matrix whose columns are the

TABLE 5

Recognition Rate, R, on Training Data Using Each of the 38 Features in the Original Feature Set

Feature	R (in %)	Feature	R (in %)	Feature	R (in %)	Feature	R (in %)
C_1	38.44	DC_1	30.85	DDC_1	20.54	DE	31.56
C_2	30.90	DC_2	29.26	DDC_2	17.23	DDE	28.28
C_3	28.90	DC_3	26.10	DDC_3	17.90		
C_4	23.33	DC_4	17.05	DDC_4	12.15		
C_5	20.00	DC_5	15.59	DDC_5	11.26		
C_6	16.69	DC_6	13.79	DDC_6	12.13		
C_7	17.13	DC_7	15.00	DDC_7	13.74		
C_8	15.90	DC_8	15.54	DDC_8	12.54		
C_9	18.90	DC_9	16.49	DDC_9	13.51		
C_{10}	19.00	DC_{10}	15.77	DDC_{10}	11.90		
C_{11}	17.95	DC_{11}	15.74	DDC_{11}	11.74		
C_{12}	15.15	DC_{12}	13.13	DDC_{12}	11.08		

eigenvectors corresponding to d largest eigenvalues of $\mathbf{T}$. Recognition results on the test data set as a function of d are shown in Table 9. It can be seen from this table that this method performs slightly worse than method 2. But, its performance is better than that of other methods. This is despite the fact that it neither maximizes the between-class separation nor minimizes within-class scatter. Also, note that the recognition performance with the 38 features in the transformed feature space is not as good as that with the 38 features in the original feature space. Again, this happens due to the use of diagonal covariance matrices in the HMM-based recognizer.

5. DISCUSSION OF RESULTS

In the preceding section, we have studied four different methods for reducing the dimensionality of the feature space. A Gaussian mixture density HMM-based speech recognizer with diagonal covariance matrices has been used for recognizing the isolated words in this study. It has been shown that the feature selection method using the recognition rate on training data as the figure of merit performs best among these methods. This method can reduce the dimensionality of the feature space to 16 without sacrificing in terms of recognition performance. (In fact, the recognition performance using these 16 features is better than that obtained by using all the 38 features in the original feature set.) There are two reasons for its success. First, use of diagonal covariance matrices in the HMM-based speech recognizer justifies the use of a figure of merit for feature selection. Second, the figure of merit used in this method directly reflects the recognition capabilities of the individual features.

Though this method has met the desired goal of reducing the dimensionality and, at the same time,

TABLE 6

Rank-Ordering of the 38 Features in the Original Feature Set Using Recognition Rate on Training Data as the Figure of Merit

Rank	Feature	Rank	Feature	Rank	Feature	Rank	Feature
1	C_1	13	C_9	25	DC_8	37	DDC_5
2	DE	14	C_{11}	26	C_{12}	38	DDC_{12}
3	C_2	15	DDC_3	27	DC_7		
4	DC_1	16	DDC_2	28	DC_6		
5	DC_2	17	C_7	29	DDC_7		
6	C_3	18	DC_4	30	DDC_9		
7	DDE	19	C_6	31	DC_{12}		
8	DC_3	20	DC_9	32	DDC_8		
9	C_4	21	C_8	33	DDC_4		
10	DDC_1	22	DC_{10}	34	DDC_6		
11	C_5	23	DC_{11}	35	DDC_{10}		
12	C_{10}	24	DC_5	36	DDC_{11}		

TABLE 7

Recognition Accuracy as a Function of the Size d of the Reduced Feature Set

Size of reduced feature set	Recognition accuracy in (%)
4	72.41
8	86.33
12	88.51
16	89.18
20	89.36
24	89.31
28	89.59
32	89.72
36	89.38
38	88.87

Note. Dimensionality reduction is done by using recognition rate on training data as the figure of merit.

TABLE 8

Recognition Accuracy as a Function of the Size d of the Reduced Feature Set

Size of reduced feature set	Recognition accuracy in (%)
4	78.03
8	86.03
12	87.15
16	87.56
20	88.21
24	88.59
28	87.87
32	88.13
36	87.72
38	87.62

Note. Dimensionality reduction is done by using the linear discriminant analysis.

not sacrificing in terms of recognition performance, the question remains whether one can justify the inclusion of certain features in the reduced feature set on some physical grounds. If this is possible, then this reduced feature set will have more appeal in the sense that it will be useful for other data bases as well as for other recognition tasks (such as connected word recognition and large-vocabulary continuous speech recognition). In order to elaborate on this point further, let us look at the rank-ordered list of features in Table 6, which has been obtained by using the recognition rate on training data as the figure of merit. Let us consider the reduced feature set of size $d = 18$. This feature set includes delta energy, delta-delta energy, the first four delta cepstral coefficients, the first three delta-delta cepstral coefficients, and the following nine cepstral coefficients: C_1, C_2, C_3, C_4, C_5, C_7, C_9, C_{10}, and C_{11}. The cepstral coefficients C_6 and C_8 are not included in this feature set, while the higher cepstral coefficients C_7, C_9, C_{10}, and C_{11} are a part of this feature set. There is no physical reason we can think of that explains this behavior. It is not clear whether an arbitrary sequence of cepstral coefficients has any spectral meaning. But, an ordered sequence of the first few (say, Q) cepstral coefficients has a spectral meaning. In order to illustrate this point, the power spectra of the three vowel sounds (/i/, /a/, and /u/)

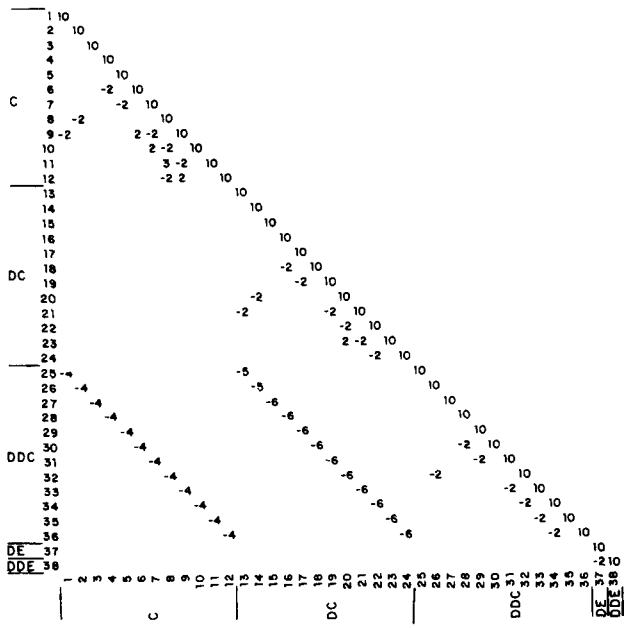

FIG. 1. Correlation (times 10) between different features in the original feature set. Correlation values between −0.2 and 0.2 are not shown.

TABLE 9

Recognition Accuracy as a Function of the Size d of the Reduced Feature Set

Size of reduced feature set	Recognition accuracy in (%)
4	75.90
8	86.85
12	88.44
16	89.44
20	89.08
24	89.44
28	88.87
32	88.59
36	87.77
38	87.87

Note. Dimensionality reduction is done by using the principal component analysis.

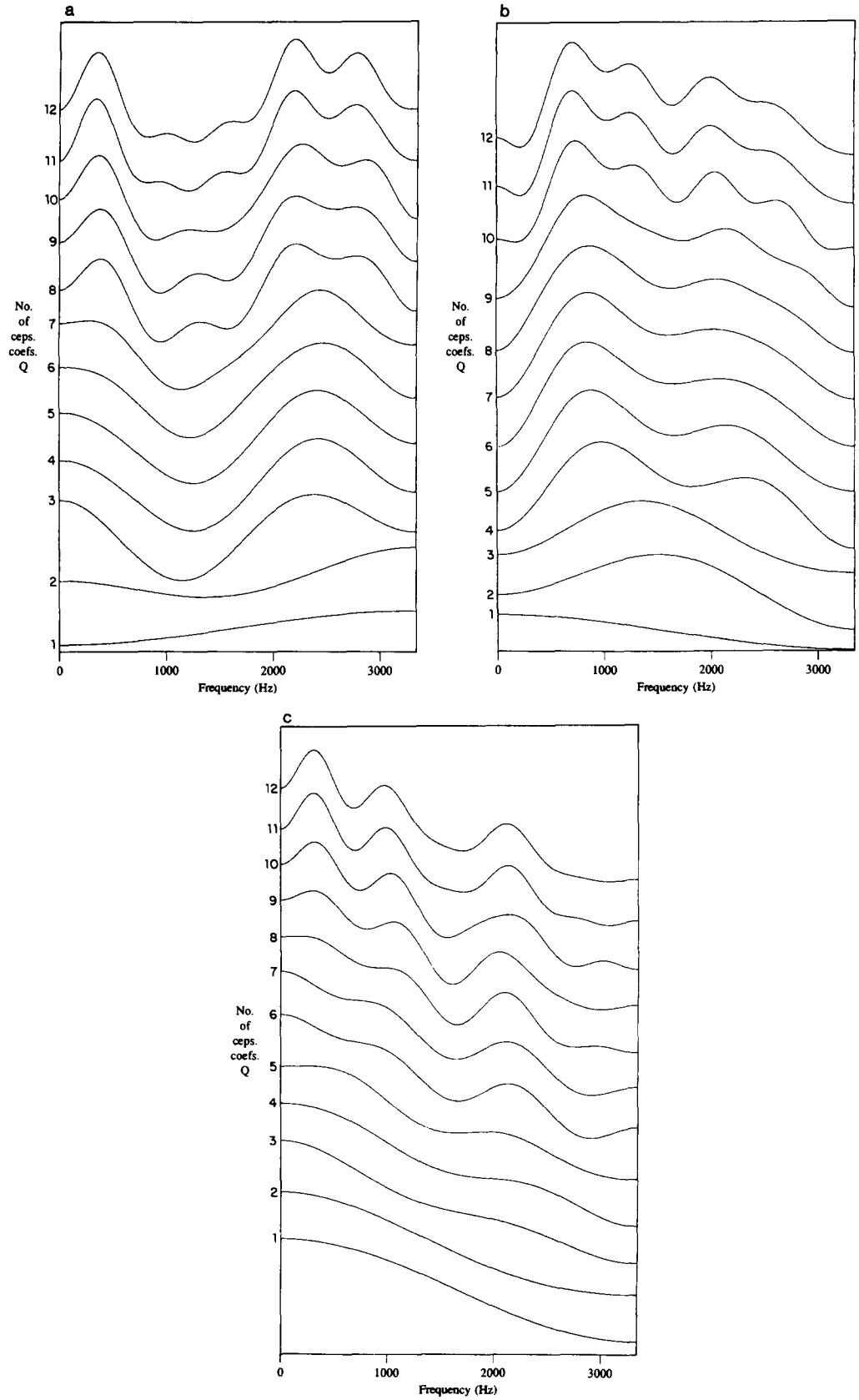

FIG. 2. Power spectrum (in dB) for different values of Q. (a) Vowel /i/, (b) vowel /a/, and (c) vowel /u/.

for different Q values are shown in Fig. 2, and the running power spectra of all the frames in a speech utterance of word /bi/ are shown in Fig. 3. We can see from these figures that the ordered sequence of the first Q cepstral coefficients has a definite meaning. By increasing the value of Q, we add more details to the power spectrum. The power spectrum corresponding to longer cepstral sequences has better resolution. Also, we can see that about 8 to 10 cepstral coefficients are sufficient to get all the details in the spectrum; i.e., we do not gain in terms of spectral resolution by increasing Q beyond 10.

Thus, if we use the first Q_0 cepstral coefficients, the first Q_1 delta cepstral coefficients, and the first Q_2 delta-delta cepstral coefficients in our feature set, this feature set will have some physical meaning. Note that we always include delta energy and delta-delta energy in our feature set. Thus, the dimensionality of the reduced feature set is $d = Q_0 + Q_1 + Q_2 + 2$. In order to find the proper values for Q_0, Q_1, and Q_2, we must know the relative importance of the cepstral coefficients, the delta cepstral coefficients, and the delta-delta cepstral coefficients for speech recognition. For this, we plot in Fig. 4 the recognition rate on training data for individual cepstral features. We can see from this figure that for speech recognition the cepstral coefficients are more important than the delta cepstral coefficients, which, in turn, are more important than the delta-delta cepstral coefficients. In other words, we have to include more cepstral coefficients in our reduced feature set than the delta cepstral coefficients, and more delta cepstral coefficients than the delta-delta cepstral coefficients. This makes sense from a spectral viewpoint. We add more details to the power spectrum by increasing the Q value, but these details may not be necessary for computing the temporal derivatives of the spectrum. In fact, these details make the temporal derivatives more noisy (as can be seen from Fig. 3). Thus, as we go to higher temporal derivatives, we should have less of these details. Since there is a direct correspondence between temporal derivatives of the log-power spectrum and temporal derivatives of cepstral coefficients [3], it follows that we should use smaller values of Q for getting a reliable estimate of higher temporal derivatives. In other words, higher is the order of the temporal derivative, smaller should be the value of Q. Thus, $Q_0 \geq Q_1 \geq Q_2$.

Using this rule of thumb, we have arrived at a number of reduced feature sets of different sizes. Recognition performance of the HMM-based speech recognizer using these reduced feature sets is evaluated on the test data set. Results are shown in Table 10. Note that the reduced feature sets investigated here include at least eight cepstral coefficients (i.e., $Q_0 \geq 8$). This is done to capture in the feature set all the details of the power spectrum. It can be seen from this table that these feature sets perform better than the reduced feature sets obtained by using the dimensionality reduction methods reported in the preceding section. Since we can justify the formation of these feature sets on physical grounds, we expect them to perform equally well on other recognition tasks and data bases. However, this has to be investigated.

Recently, Bocchieri and Wilpon [28] from our laboratories have investigated the dimensionality reduction problem for the same feature set (that has been studied in the present paper), and using a similar HMM-based speech recognizer. They have investigated a feature selection method, where they use a figure of merit which tries to maximize the discrimination between the classes. Using their reduced feature sets, we have conducted recognition experiments on

TABLE 10

Recognition Accuracy as a Function of the Size d of the Reduced Feature Set Which Is Derived from Physical Considerations

Size of reduced feature set $d(Q_0 + Q_1 + Q_2 + 2)$	Recognition accuracy in (%)
16(8 + 3 + 3 + 2)	89.51
18(8 + 5 + 3 + 2)	89.95
20(8 + 7 + 3 + 2)	90.15
22(9 + 8 + 3 + 2)	90.26
24(10 + 8 + 4 + 2)	90.41

Note. In addition to delta energy and delta–delta energy, the reduced feature set has the first Q_0 cepstral coefficients, the first Q_1 delta cepstral coefficients, and the first Q_2 delta–delta cepstral coefficients, where $Q_0 \geq Q_1 \geq Q_2$.

TABLE 11

Recognition Performance Obtained by Using Bocchieri and Wilpon's Reduced Feature Set

Size of reduced feature set	Recognition accuracy in (%)
4	56.41
8	78.31
12	87.18
16	88.87
20	89.72
24	89.69
28	89.85
32	90.03
36	89.18
38	88.87

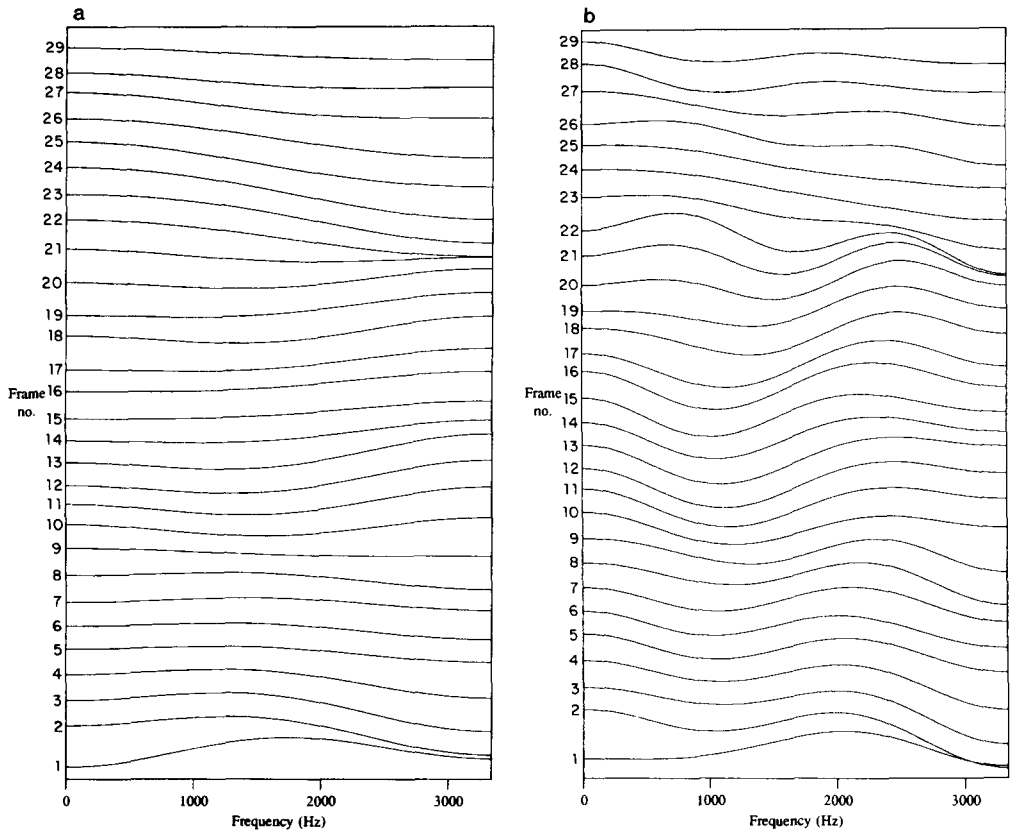

FIG. 3. Running power spectra (in dB) of all the frames in the speech utterance /bi/. (a) $Q = 2$, (b) $Q = 4$, (c) $Q = 6$, (d) $Q = 8$, (e) $Q = 10$, and (f) $Q = 12$.

our isolated word recognition task. Results are shown in Table 11. It can be seen from this table that their dimensionality reduction method performs as well as the dimensionality reduction methods used in our study (see Tables 4, 7, 8, and 9). However, comparison of Tables 10 and 11 shows that the reduced feature sets derived in the present paper from physical considerations perform better than Bocchieri and Wilpon's reduced feature sets.

6. CONCLUSIONS

In this paper, the problem of dimensionality reduction is studied for the enhanced feature set used in the speech recognizers at AT&T Bell Laboratories. This feature set consists of the following 38 features: 12 cepstral coefficients, 12 delta cepstral coefficients, 12 delta-delta cepstral coefficients, delta energy, and delta-delta energy. The speech recognizer employed in this study uses continuous density HMMs with Gaussian mixture densities, where the covariance matrices are assumed to be diagonal. Four different methods of dimensionality reduction have been investigated. The dimensionality reduction method that uses the recognition rate on training data as the figure of merit for feature selection has been found to give the best performance. Though this method achieves the desired goal of reducing the dimensionality of the feature space to 16 and, at the same time, not sacrificing in terms of recognition performance, some of the features that are selected by this method are rather arbitrary and cannot be justified on physical grounds. In order to ensure that the features in the reduced feature set have physical meaning, it has been proposed that this feature set should include, in addition to delta energy and delta-delta energy, the first Q_0 cepstral coefficients, the first Q_1 delta cepstral coefficients, and the first Q_2 delta-delta cepstral coefficients, where $Q_0 \geq Q_1 \geq Q_2$. This reduced feature set has been found to result in better performance than the reduced feature sets obtained by different dimensionality reduction methods investigated in this paper.

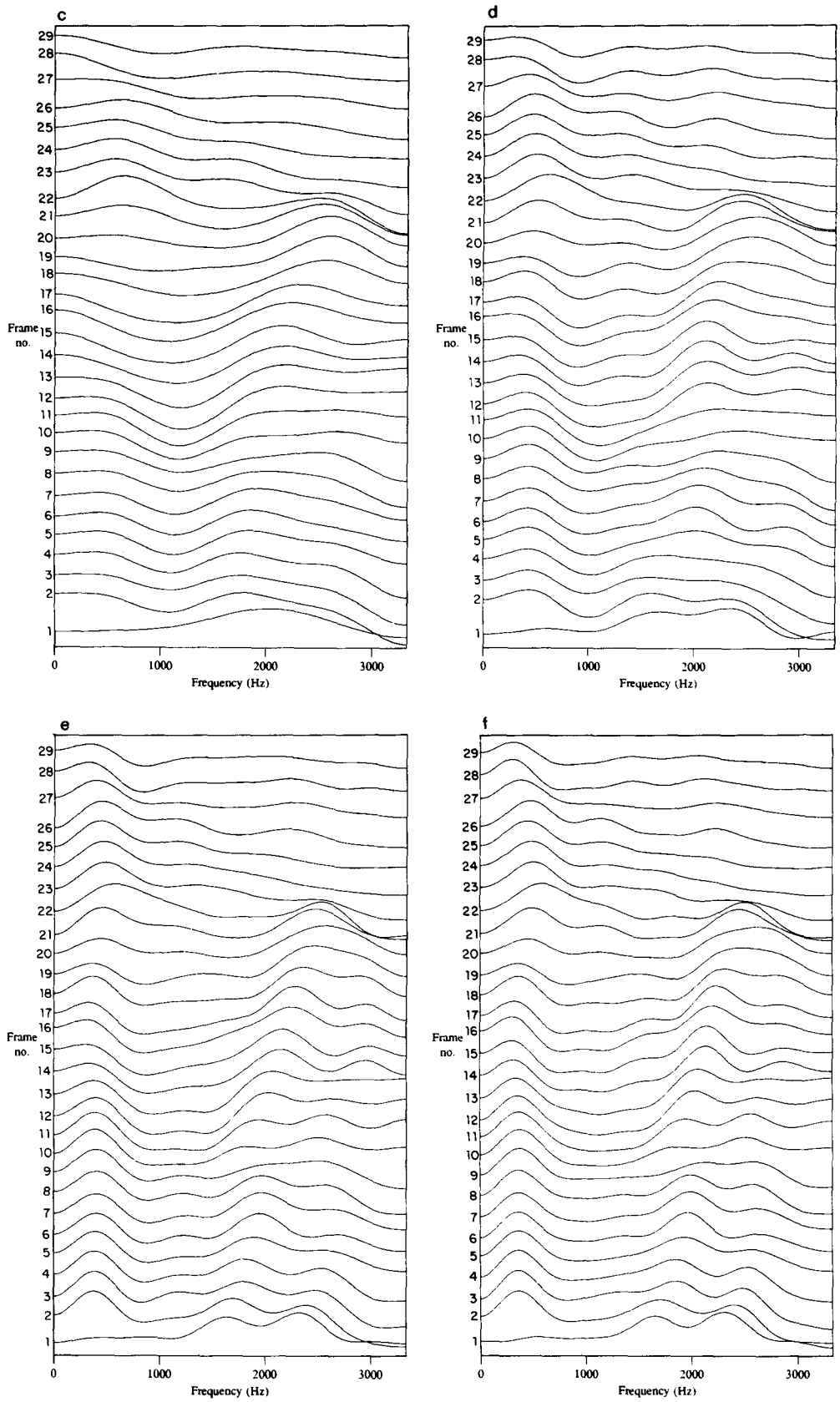

FIG. 3—*Continued*

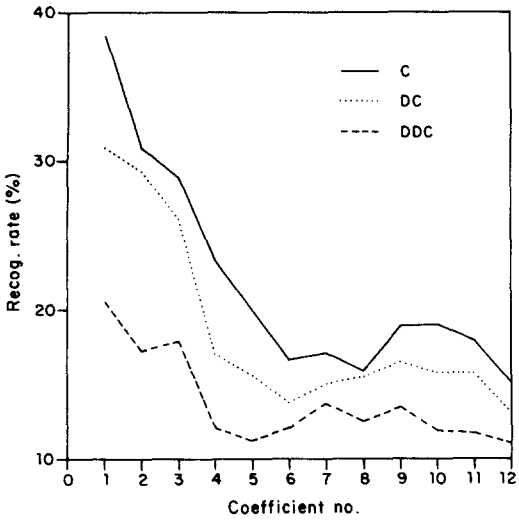

FIG. 4. Recognition rate on training data using individual cepstral features. Here, C stands for the cepstral coefficients, DC for the delta cepstral coefficients, and DDC for the delta-delta cepstral coefficients.

ACKNOWLEDGMENTS

The author is thankful to B. S. Atal and M. M. Sondhi for useful discussions during the course of this work.

REFERENCES

1. Rabiner, L. R., Wilpon, J. G., and Juang, B. H. A model-based connected-digit recognition system using either hidden Markov models or templates. *Comput. Speech Language* **1** (1986), 167–197.
2. Furui, S. Speaker-independent isolated word recognition using dynamic features of speech recognition. *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-34** (Feb. 1986), 52–59.
3. Furui, S. Speaker-independent isolated word recognition based on emphasized spectral dynamics. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Tokyo, Japan,* Apr. 1986, pp. 1991–1994.
4. Rabiner, L. R., Wilpon, J. G., and Soong, F. K. High performance connected digit recognition using hidden Markov models. *IEEE Trans. Acoust. Speech Signal Process.* **37** (Aug. 1989), 1214–1225.
5. Lee, K. F. Large-vocabulary speaker-independent continuous speech recognition: The SPHINX system. Ph.D. thesis, Computer Science Department, Carnegie Mellon University, Pittsburgh, PA, 1988.
6. Lee, C. H., Rabiner, L. R., Pieraccini, R., and Wilpon, J. G. Acoustic modeling for large vocabulary speech recognition. *Comput. Speech Language* **4** (Apr. 1990), 127–165.
7. Hanson, B. A., and Applebaum, T. H. Robust speaker-independent word recognition using static, dynamic and acceleration features: Experiments with lombard and noisy speech. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Albuquerque, NM,* Apr. 1990, pp. 857–860.
8. Ney, H. Experiments on mixture-density phoneme modeling for the speaker independent 1000-word speech recognition DARPA task. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Albuquerque, NM,* Apr. 1990, pp. 713–716.
9. Hanson, B. A., and Applebaum, T. H. Features for noise-robust speaker-independent word recognition. In *Proc. Int. Conf. Spoken Language Processing, Kobe, Japan,* Nov. 1990, pp. 1117–1120.
10. Furui, S. Cepstral analysis techniques for automatic speaker verification. *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-29** (Apr. 1981), 254–272.
11. Soong, F. K., and Rosenberg, A. E. On the use of instantaneous and transitional spectral information in speaker recognition. *IEEE Trans. Acoust. Speech Signal Process.* **36** (June 1988), 871–879.
12. Mason, J. S., and Zhang, X. Velocity and acceleration features in speaker recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Toronto, Canada,* May 1991, pp. 3673–3676.
13. Wilpon, J. G., Lee, C. H., and Rabiner, L. R. Improvements in connected digit recognition using higher order spectral and energy features. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Toronto, Canada,* May 1991, pp. 349–352.
14. Lee, C. H., Giachin, E., Rabiner, L. R., Pieraccini, R., and Rosenberg, A. E. Improved acoustic modeling for speaker independent large vocabulary continuous speech recognition. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Toronto, Canada,* May 1991, pp. 161–164.
15. Huang, X. D., Lee, K. F., Hon, H. W., and Hwang, M. Y. Improved acoustic modeling with the SPHINX speech recognition system. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Toronto, Canada,* May 1991, pp. 345–348.
16. Kanal, L. N., and Chandrasekaran, B. On dimensionality and sample size in statistical pattern classification. *Pattern Recognition* **3** (Oct. 1971), 225–234.
17. Devijver, P. A., and Kittler, J. *Pattern Recognition: A Statistical Approach.* Prentice-Hall, Englewood Cliffs, NJ, 1982.
18. Pruzansky, S. Talker-recognition procedure based on analysis of variance. *J. Acoust. Soc. Am.* **36** (Nov. 1964), 2041–2047.
19. Das, S. K., and Mohn, W. S. A scheme for speech processing in automatic speaker verification. *IEEE Trans. Audio Electroacoust.* **AU-19** (Mar. 1971), 32–43.
20. Bricker, P. D., Gnanadesikan, P., Mathews, M. V., Pruzansky, S., Tukey, P. A., Wachter, K. W., and Warner, J. L. Statistical techniques for talker identification. *Bell Syst. Tech. J.* **50** (1971), 1427–1454.
21. Mohn, W. S., Jr. Two statistical feature evaluation techniques applied to speaker identification. *IEEE Trans. Comput.* **C-20** (Sept. 1971), 979–987.
22. Wolf, J. J. Efficient acoustic parameters for speaker recognition. *J. Acoust. Soc. Am.* **51** (1972), 2044–2056.
23. Atal, B. S. Automatic speaker recognition based on pitch contours. *J. Acoust. Soc. Am.* **52** (1972), 1687–1697.
24. Sambur, M. R. Selection of acoustic features for speaker identification. *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-23** (Apr. 1975), 176–182.
25. Bahl, L. R., Brown, P. F., de Souza, P. V., and Mercer, R. L. Speech recognition with continuous parameter hidden Markov models. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, New York, NY,* Apr. 1988, pp. 40–43.
26. Hunt, M. J., and Lefebvre, C. A comparison of several acoustic representations for speech recognition with degraded and undegraded speech. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Glasgow, Scotland,* May 1989, pp. 262–265.
27. Hunt, M. J., Richardson, S. M., Bateman, D. C., and Piau, A. An investigation of PLP and IMELDA acoustic representations and of their potential for combination. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Toronto, Canada,* May 1991, pp. 881–884.

28. Bocchieri, E. L., and Wilpon, J. G. Discriminative analysis for feature extraction in automatic speech recognition. Internal Technical Memorandum, AT&T Bell Laboratories, Oct. 1991.
29. Chan, W. W. Enhanced features for connected digit recognition. M.S. thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology, Cambridge, MA, Feb. 1992.
30. Wilks, S. S. *Mathematical Statistics.* Wiley, New York, 1962.
31. Markel, J. D., and Gray, A. H., Jr. *Linear Prediction of Speech.* Springer-Verlag, Berlin, 1976.
32. Juang, B. H., Rabiner, L. R., and Wilpon, J. G. On the use of bandpass liftering in speech recognition. *IEEE Trans. Acoust. Speech Signal Process.* **ASSP-35** (1987), 947–954.
33. Rabiner, L. R. A tutorial on hidden Markov models and selected applications in speech recognition. *Proc. IEEE* **77** (Feb. 1989), 257–286.
34. Linde, Y., Buzo, A., and Gray, R. M. An algorithm for vector quantizer design. *IEEE Trans. Commun.* **COM-28** (Jan. 1980), 84–95.
35. Huang, X., and Jack, M. Semi-continuous hidden Markov models for speech signals. *Comput. Speech Language* **3** (July 1989), 239–251.
36. Huang, E. F., and Soong, F. K. A probabilistic acoustic map based discriminative HMM training. In *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing, Albuquerque, NM,* Apr. 1990, pp. 693–696.
37. Bellegarda, J. R., and Nahamoo, D. Tied mixture continuous parameter modeling for speech recognition. *IEEE Trans. Acoust. Speech Signal Process.* **38** (Dec. 1990), 2033–2045.
38. Rabiner, L. R., and Wilpon, J. G. Some performance benchmarks for isolated word speech recognition systems. *Comput. Speech Language* **2** (Dec. 1987), 343–357.

KULDIP K. PALIWAL was born in Aligarh, India, in 1952. He received the B.S. degree from Agra University, India, in 1969; the M.S. degree from Aligarh University, India, in 1971; and the Ph.D. degree from Bombay University, India, in 1978. Since August 1972, he has been with Tata Institute of Fundamental Research, Bombay, India, where he has worked on various aspects of speech processing, e.g., speech recognition, speech coding, and speech enhancement. From September 1982 to October 1984, he was an NTNF fellow at the Department of Electrical and Computer Engineering, Norwegian Institute of Technology, Trondheim, Norway. He was a visiting scientist at the Department of Communications and Neuroscience, University of Keele, United Kingdom, during June–September 1982 and January–March 1984, and at the Electronics Research Laboratory (ELAB), Norwegian Institute of Technology, Trondheim, Norway, during April–July 1987, April–July 1988, and March–May 1989. From May 1989 to December 1991, he was at the Acoustics and Speech Research Departments, AT&T Bell Laboratories, Murray Hill, New Jersey. His work has been concentrated on quantization of linear predictive coding parameters, fast search algorithms for vector quantization, feature analysis and distance measures for speech recognition, and robust spectral analysis techniques. His current research interests are directed toward automatic speech recognition using hidden Markov models and neural networks. He is a Fellow of the Acoustical Society of India. He is a member of the IEEE Signal Processing Society's Technical Committee on Neural Networks.