Speech Recognition Using Linear Prediction Residue and a Reduced Feature Space

F. Mavaddat and S.K.S. Cheng

Department of Computer Science
University of Waterloo
Waterloo, Ontario, Canada

Research Report CS-82-34

December, 1982

Speech Recognition Using Linear Prediction Residue and a Reduced Feature Space

F. Mavaddat
S.K.S. Cheng
CS-82-34

Department of Computer Science
University of Waterloo
Waterloo, Ontario N2L 3G1
Canada

ABSTRACT

Label Space is defined as a space to which reference points of a feature space can be mapped. Label space redundancies are removed by eliminating points from the feature space. It is shown that some similarity measurements can be performed in a combination of feature and label space with computational advantages over the conventional methods. It is further shown that the similarity measurements between the signals of speech in the space of LPC features can take benefit of these properties and a new algorithm for word similarity studies is proposed. Two experiments are performed for finding a near optimum set of parameters for the system and measuring its power for recognition of isolated words. Some thoughts on hardware implementation is also given.

Keywords: linear predictive coding, feature space, similarity measurements, isolated word recognition, phonemic labelling.

Speech Recognition Using Linear Prediction Residue and a Reduced Feature Space

F. Mavaddat

S.K.S. Cheng

CS-82-34

Department of Computer Science

University of Waterloo

Waterloo, Ontario N2L 3G1

Canada

Introduction

There are three steps to every pattern recognition problem namely, extraction of features, measuring of similarities, and labelling of the unknown input. Theoretically, each step depends only on the outcome of the previous step. Practically, there are strong feedback considerations within these steps and the merit of each step can be judged only within the framework of the total system. This is the view which should be held in the study of this paper.

In this paper we first demonstrate a coding technique for transformation of reference points, from feature space, into codewords. Such codewords are representable as points in the label space. We will then show that the similarity measurements in the feature space are also measurable by the codeword distances in the label space. Such codewords often possess a certain amount of redundancy which can be used towards reducing the cost of the computations.

Some of this redundancy can be removed through omission of some reference points from the feature space. Such reduced feature spaces are then studied in relation to linear predictive

coding of speech signals. It is shown that in addition to reducing the number of reference templates in the LP feature space, computationally less expensive similarity measurements can also be employed.

Based on these an acoustic processor is proposed. The acoustic processor will assign codewords to suitable intervals of speech. By concatenation of these codewords, label matrices are formed. Utterances are compared by measuring similarity of corresponding label matrices.

An experimental study is made and it is shown that the proposed techniques possess considerable resolving power.

1. The Label Space and the Similarity Measurements

Traditionally, patterns of short-time speech signals have been represented by feature vectors. Such feature vectors are made of formalized entities representing the relevant parameters, while ignoring redundancies, unrelated data, and noise. There are probably as many feature vectors proposed as there are systems designed for processing of speech. Some of them, representing distinct concepts, can be found in [Broad], [Atal1],and [Fujiskai].

Independent of their underlying concepts, such feature vectors are usually representable as points in their corresponding feature space. Similarity of patterns are measured in terms of suitable distance measures between the reference points in the feature space.

1.1 The Label Space

Let us consider n reference patterns by their respective points; $R(1)$, $R(2)$,...,$R(n)$ in the feature space. We also consider the distance measure $d(i,j)$ to be a similarity measure between the *ith* and the *jth* reference points, where $d(i,i) = 0$ for $1 \leq i \leq n$, and $d(i,j) > 0$ for $1 \leq i \leq n$, $1 \leq j \leq n$, and $i \neq j$. We will further assume that a unique label is associated with each of the reference patterns.

Corresponding to each $R(i)$ we will define a codeword, $L(i)$, as an ordered vector of reference point labels, such that $R(j)$'s label will precede that of $R(k)$ in $L(i)$, if $d(i,j) < d(i,k)$. $L(i)$ will be called the codeword of the *ith* pattern. Codewords corresponding to individual patterns are always unique by having their own label as their first element.

Fig. (1) is an example of a feature space with eight reference points labelled as a,b,c,d,e,f,g,and h. Distances between these reference points are shown in Fig. (1a). Figure (1b) shows the reference points as defined by their codewords.

We define the distance $D(i,j)$ between $L(i)$ and $L(j)$ in the label space by

$$D(i,j) = \sum_{k=1}^{n} |(p(i,k) - p(j,k))| \qquad (1)$$

(a)

	a	b	c	d	e	f	g	h
a	0	2	3	1	4	6	7	5
b	2	0	1	3	6	4	5	7
c	3	1	0	2	7	5	4	6
d	1	3	2	0	5	7	6	4
e	4	6	7	5	0	2	3	1
f	6	4	5	7	2	0	1	3
g	7	5	4	6	3	1	0	2
h	5	7	6	4	1	3	2	0

(b)

L(1) = adbcehfg
L(2) = bcadfgeh
L(3) = cbdagfhe
L(4) = dacbhegf
L(5) = ehfgadbc
L(6) = fgehbcad
L(7) = gfhecbda
L(8) = hegfdacb

Fig. (1): Distance measures of a hypothetical system in the feature space (a) and their corresponding codewords (b).

where $p(i,k)$ $(p(j,k))$ represents the index position of the *kth* label in $L(i)$ $(L(j))$ vector and n is the number of label points in the feature space.

1.2 Similarity Measurements in the Label Space

Recognition is the process of associating the unknown input pattern with one of the known reference patterns through some similarity measurements. Traditionally this has been done by measuring the similarity of the unknown pattern with each of the known patterns through distance measurements in the space of derived features. We will now show that this decision can also be based on similarity measurements in the label space. This new similarity measurement can sometimes have computational advantages compared to those made in the feature space. In section 2- of this paper one such possible advantage is studied in relation to the traditional measurements in the linear prediction feature space.

Measuring similarities in the label space has two phases. During the first phase the distance $d(x,i)$ between the feature vector of the unknown input, $R(x)$, and all reference points, $R(i)$ for $1 \leq i \leq n$ are measured. Based on these measurements $L(x)$ is formed and recognition is based on measuring the distance $D(x,i)$ between the $L(x)$ and all other $L(i)$s for $1 \leq i \leq m$. Traditional decision algorithms can then be applied to the association of x with one of the reference patterns.

Nearest neighbour algorithm, is a special case of the above more generalized algorithm which, in a sense, expects an exact match between one of the known and the unknown codewords. Because of the unique representability of the codewords by their first element, this exact match

can be reduced to that of comparing the first elements. This eliminates the need for the formation of the codewords and the two are considered matching if their first elements correspond. Labelling of the unknown input by its first codeword element, its nearest neighbour, is called the "nearest neighbour labelling algorithm".

Codewords posses a good amount of redundancy. Nearest neighbour labelling algorithm is one way of using this redundancy towards reducing the amount of computation. Another possibility is through reducing the number of reference points in the feature space. Following is a discussion of ways of removing this type of redundancy.

Let us consider n codewords $L(1)$, $L(2)$,...,$L(n)$ each made from some permutations of n distinct labels. These words are all M-distinct if

$$D(i,j) > M \quad 1 \leq i \leq n, \quad 1 \leq j \leq n, \quad i \neq j \tag{2}$$

where $D(i,j)$ is defined by (1) and M is some non negative integer. A reference point, in the feature space, is said to be M-removable (or simply "removeable") if after omitting its label from all codewords they remain M-distinct (distinct).

Intuitively one expects that m of such reference points are "removable" if $n \ll (n-m)!$. This need is usually satisfied by the typical values of n and m in many real systems. Fig. (2) shows the codewords corresponding to the eight reference points of Fig. (1) after removal of one, two, and three labels from the feature space.

	I	II	III
L(1)=	bdcehfg	dcehfg	dehfg
L(2)=	bcdfgeh	cdfgeh	dfgeh
L(3)=	cbdgfhe	cdgfhe	dgfhe
L(4)=	dcbhegf	dchegf	dhegf
L(5)=	ehfgdbc	ehfgdc	ehfgd
L(6)=	fgehbcd	fgehcd	fgehd
L(7)=	gfhecbd	gfhecd	gfhed
L(8)=	hegfdcb	hegfdc	hegfd

Fig. (2): The codewords of Fig. (1) after removing (I) the label "a", (II) the labels "a" and "b", (III) the labels "a", "b", and "c".

Fig. (2) shows that for associating an unknown input pattern with one of the known pat-

terns, it is not necessary to compare it with all of the eight patterns in the feature space. This in addition to savings in the storage requirements, may have computational advantages if similarity studies in the feature space are costlier than those in the label space, which is sometimes the case. In general this reduction in the number of points in the feature space can be computationally advantageous if

$$\frac{n}{m} < \frac{c(f)}{c(l)} \tag{3}$$

where $c(f)$ and $c(l)$ are the computational costs of the similarity studies in the feature and the label space respectively, n is the total number of the reference points in the feature space, and m $(m<n)$ is the number of the redundant reference points to be removed. Obviously there will be no advantage if $c(f) \leq c(l)$.

Later in this paper we will see that the $c(f)$ associated with the first phase of the two phase algorithm applied to the linear prediction residual, as the basis of similarity measurements, is less expensive compared to those required to be performed totally in the feature space. This will further add to the computational advantage of the proposed algorithm.

Study of the procedures for systematic removal of the redundant reference points from the feature space is not of our immediate interest. At worst it can be based on an exhaustive search algorithm subject to certain rules for speeding up of the operations. Being executed once for the lifetime of each system its overhead should be tolerable.

In some pattern recognition problems prior knowledge of all reference points is not feasible. This can be due to their large number and/or great variations. Phonemic labelling is an example of this possibility [Jelinek] , [Reddy]. Under such conditions, and if the absolute uniqueness of label vectors of all patterns is not immediately related to the overall performance of the system, reference points can be chosen without the help of a solid redundancy removal algorithm.

Following points should be considered in the choice of such points:

1- The number of remaining reference points, $(n-m)$, should satisfy the $n \ll (n-m)!$ relation.

2- Remaining reference points should statistically be prominant within the input population.

3- The selected features should have a high yield in the lexicon of the input words [Shoup].

2. The Label Space of LP Features

Here we will apply the two phase detection algorithm to the study of pattern similarities in the linear prediction feature space. The main reason for consideration of the linear prediction feature space is its wide acceptance in modelling of speech signals [Atal1], [Atal2], [Makhoul] and recognition of speech [Itakura], [White], [Coker], [Gupta].

Questions regarding pattern similarities can be asked in two different ways, leading to distinct formulations with computationally significant differences. The first type of the questions deal with similarities, between the two patterns, in absolute terms. Answers to such questions are in terms of distance measures signifying, great, little, or no similarity. The second type of the questions deal with the similarity of a given pattern with two or more others in relative terms. These questions are in fact asking for ordering of the many according to their similarity with one. It is not difficult to see that the questions of the first type are more demanding and can always be used in answering the second type, while the reverse is not true. We will later show that the answer to the questions of the first type also require computationally more complex and expensive calculation in the feature space of the LP coefficients.

Studies made and techniques proposed so far for measuring the similarity of speech patterns in the space of LP features deal with the questions of the first type. This is understandable in view of the eventual use which is made of them for comparing the two short intervals of speech taken from the unknown and the known patterns according to some suitable time wrapping function. On the other hand, the first phase of measuring the similarities of two intervals in the label space requires only the answer to questions of the second type. Here, we will first study some of the proposed distance measures in the LP feature space and then investigate the ways that they

can be fit to coding of label vectors.

There has been a considerable amount of interest in the study of a suitable distance measure, based on features derived by LP techniques, over the last few years [Itakura], [Coker], [Souza], [Gupta], [Tribolt]. Direct comparison of LP coefficients is shown to be unsatisfactory [Sato]. Itakura attributes this "to the fact that the feature space spanned by LPC is too complicated to introduce a simple and effective measure of distance between elements" [Itakura]. It turns out that all successful formulations, in one way or the other, are based on some form of the power of the residual signal obtained by filtering of one by the inverse model of the other. Here we will study some of these from a general point of view and consider the ways that they can be applied to coding of label vectors.

It is generally accepted that a short interval of speech signal, $X(n)$, can be considered to be stationary. It has been further shown that, except for nasals, such stationary intervals of speech signal can be modelled by an all pole model [Flannagan], [Makhoul]

$$X(n) = -\sum_{k=1}^{p} a(k)\,.X(n-k) + e\,(n) \tag{4}$$

where $e(n)$ is a white noise process and the values of $a(k)$ for $1 \leq k \leq p$, can be obtained by solving the following system of normal equations

$$\sum_{k=1}^{p} a(k)\,.R(i-k) = -R(i) \quad 1 \leq i \leq p \tag{5}$$

where $R(i) = R(-i)$ is the autocorrelation function of the signal $X(n)$. The vector $\overline{a} = (1, a(1), a(2), \ldots a(p))$ whose $a(k)$ for $1 \leq k \leq p$, values are obtainable from (5) will be called the linear prediction model, or simply the model, of the signal $X(n)$.

The so called residue vector $\overline{e}^{\,T} = (e(0),\ e(i),\ e(2), \ldots e(p))$ can be defined as

$$\overline{e} = R\overline{a}^{T} \tag{6}$$

where R is the pth order autocorrelation matrix of an arbitrary signal $X1(n)$, and $\overline{a} = (1,\ a(1),\ a(2), \ldots a(p))$ is the model of another (or the same) signal, $X2(n)$ (or $X1(n)$). Different functions of $\overline{e}$, $F(\overline{e}\,)$, have the following useful properties in measuring the similarities of the two signal $X1(n)$ and $X2(n)$.

1- Should $\bar{a}$ be a model of itself

$$F(\bar{e}) = R(0) + \sum_{k=1}^{p} a(k) * R(k) \tag{7}$$

will be the minimum total squared error of $e(n)$.

2- $F(\bar{e}) = \bar{a}^T \bar{e}$ results in the total squared error $e(n)$ due to filtering of $X1(n)$ by model $\bar{a}$ of $X2(n)$.

3- $e(i)$ for $1 \leq i \leq p$, values have been used successfully as measures of distance between signals $X1(n)$ and $X2(n)$ [Gupta] in the forms of

$$D(X1,X2) = F(\bar{e}) = \log\left(\sum_{i=1}^{p} |e(i)|\right) \tag{8}$$

or

$$D(X1,X2) = F(\bar{e}) = \log\left(\sum_{i=1}^{p} (e(i))^2\right) \tag{9}$$

where $\bar{e}$ is the residual vector of filtering $X1$ by the model of $X2$

Direct use of $F(\bar{e})$ towards answering the similrity questions of the first type should not be satisfactory. This is due to the fact that even though model $\bar{a}$ of signal $X1(n)$ results in minimum prediction error for $X1$, there is no guarantee that the same model will not result in smaller absolute residual value while filtering some other signals.

To overcome this difficulty one has to consider any $F(\bar{e})$ relative to the residual vector, say $F(\hat{e})$ which can be obtained through filtering of $X(n)$ by its own model. $F(\hat{e})$ is the self referencing component of the measurement. Should $\bar{a}$ be a model of the similar signal, then $F(\bar{e})$ and $F(\hat{e})$ are close and their ratio nears the unit value.

Under all other conditions $F(\bar{e}) > F(\hat{e})$ and this results in greater than one ratios.

This fact has been considered by a number of researchers. Coker and Bell [Coker] use $F(\bar{e}) = \bar{a}^T \bar{e}$ as the basis of their studies and propose

$$d(X,\bar{a}) = \frac{\bar{a}^T R \bar{a}}{\hat{a}^T R \hat{a}} \tag{10}$$

as a measure of similarity between X and $\overline{a}$, where $\hat{a}$ is the model of X, and R is the pth order autocorrelation matrix of X.

Itakura [Itakura], using the log likelihood ratio derives

$$d(X,\overline{a}) = \log \frac{\overline{a}^T R \overline{a}}{\hat{a}^T R \hat{a}} \tag{11}$$

as the measure of similarity between X and the model $\overline{a}$ of the reference points. It is interesting to find that Itakura's assumption of speech signals as a Gaussian random process and interpretation of similarity as a likelihood ratio leads basically to identical formulation of the distance measure derived by others purely on the basis of intuitive reasoning.

There has been some criticism of Itakura's measure on the basis of its assumed statistical properties [Souza], though Tribolt [Tribolt] has argued that such criticism is unjustified. All these measures, including those which lack the required self referencing component, have been used with success in the development of some systems.

Derivation of the input signal model, corresponding to the successive intervals of the unknown pattern, as required by (10) and (11), is one of the reasons behind the computational expense of answering the questions of the first type.

Answering the similarity questions of the second type, i.e. ordering of a set of reference patterns according to their similarity to a given input pattern, can be considerably less expensive. Suppose in answering of the questions of the second type, we measure the similarity of the input pattern with each of the reference patterns. As the denominator of the distance ratios are identical in all these measurements, and one is interested only in the relative value of these distances, it is possible to eliminate the self referencing component from all calculations (also log extraction from Itakura's measure) and base the ordering on the value of the numerators only. This should save a considerable amount of computation in formation of the label space codewords. This approach is the basis of the experimental system explained next.

3. The Experimental System

The first set of experiments deals with the search for optimum system parameters. It was performed in an informal way through which a set of reasonably performing parameters were selected.

In the second experiment, the system parameters were fixed at those values and the systems recognition rate for a vocabulary of 10 words (the ten digits) was found to be excellent.

The central idea around which the IWR is designed is that of first segmenting every utterance into an equal number of segments, each short enough to be considered stationary. Then, the similarity of every such interval with a predefined set of signals is measured and the interval is replaced by a codeword representing the ordered sequence of the known signals labels. Ordering is according to the distance measured between the interval of the unknown signal and the known ones. This leads to representing every utterance by a matrix of labels. The overall utterance similarities are measured by measuring the similarities of the label matrices.

3.1 IWR System Organization

To operate the system, for a new speaker and/or new set of words, one has to go through three phases. They are for: setting up of reference points in the feature space of phonemes, forming the reference matrices of the vocabulary, and inputting of the unknown word for measuring its similarity with the stored vocabulary.

3.1.1 Feature Space of Phonemes

Ideally one would like to have one reference point for every sound of the language. On the other hand the number of such reference points, when all allophones of every phoneme are considered, can be very large.

In section (1-2) we discussed a technique for reducing the number of such reference points through a mapping from the feature space into the label space. Such mapping can now be employed by consideration of only a limited number of phonemes in the feature space with a label

(a single letter of alphabet) assigned to everyone of them. Every other phoneme and their variations (including those already present in the feature space) can now be coded into a codeword (string of labels), with the position of label in the codeword signifying the relative distance of the coded phoneme from the phoneme in the feature space.

The question of which and how many essential phonemes should be used is a difficult one. One part of the first set of experiments is an attempt at recognition of one such set. We adopted a very informal and intuitive approach based on the guidelines discussed earlier, and experimented with a few distinct sets of phonemes.

It was intuitively obvious (and experimentally confirmed) that inclusion of vowels in the set of feature space phonemes is very helpful. Other than this, depending on the set of vocabularly, other frequently used phonemes can be included in the set. As the cost of computation directly increases by the number of such points, the tendency should be towards that of an optimum set which, with a minimum number of members, is able to produce good recognition results.

When feature space phonemes are known, one word of vocabulary for everyone should be selected (there is little point in selecting a phoneme if it does not appear in any of the words of vocabulary). This, is followed by inputting that word into the system and inspecting the utterance signal on a graphic display. This is followed by manually isolating an integer number of pitch periods from the section representing the phoneme, and deriving its model using the methods discussed in section (2). The set of all models, derived for every one of the feature space phonemes, constitutes the set of reference points in the reduced space of LP features. This phase can be interpreted as introducing a new speaker to the system, though in principle there is little against using the same model for another speaker. In hardware implementation of the system, to be discussed later, this phase corresponds to that of tuning the prediction filters to that of a particular set of sounds.

3.1.2 Introducing the Words

Once the essential phonemes are known and their models are derived and stored in the system, words to be recognized must be introduced to the system. To form a referene template for every word, several inputs of the same word should be used and "averaged" [RABINER]. We will discuss only the information of reference template based on a single utterance.

The input signal must first be sectioned into a number of segments, each equal in time and short enough to represent an allophone. As the amount of computation increases by the number of segments, the optimum number of such segments was another parameter of interest during the first set of experiments.

Once the utterance is segmented, the relative distance of every one of its segments from the feature space phonemes are calculated and a codeword representing the section is formed. The words' reference template, a matrix of labels, is formed by side concatenating the codewords representing the segments.

3.1.3 Input of the Unknown Utterance

Unknown utterances are initially processed like the known ones (3.1.2.) and a reference templates for them is formed. To recognize the unknown word its codeword matrix must be compared with that of all the known words reference templates. The measure of similarity between the unknown matrix and the known ones can be based on different measures. In our experiments we used the sum of distances between individual codewords, using a suitable time-wrapping algorithm [Sakoe] and the nearest neighbour algorithm for labelling of the unknown utterance.

The measure used to calculte the distance (dissimilarity) between the *ith* and the *jth* columns (codewords) of two label matrices is that of (1).

3.2 System Hardware

A Z80 based microcomputer system, with 64K of RAM and 4MHZ system clock is used. Peripherals include two single-density, single-sided, 8 inch floppy disk drives, a tektronix 4014

graphics terminal and the usual video terminal and the dot matrix printer.

For speech input, a close-talking head-mounted microphone is used. Input signal is bandpassed to that of telephone quality, sampled at 10KHZ, and digitized to 8 bits. All utterances were made in a small room with the low humming sound of the system cooling fans as the background noise.

3.3 The First Experiment

The first experiment is aimed at finding the optimal parameters for the system operation. The parameters under consideration were the order of prediction, the order of segmentation, and the suitable set of reference phonemes. While the first two parameters are suited to systematic search, the third has a very large space of possibilities and no systematic search was feasible. Therefore, an intuitive approach was used and a reasonable set was found. All the utterances in the first experiment were spoken by a male with a low voice whose mother tongue is English.

The vocabulary, Table (1), was taken from the first thirty words of a flight reservation system [Levinson].

YES	SUNDAY	EIGHT	MIDNIGHT
NO	ONE	NINE	ANY
MONDAY	TWO	TEN	EITHER
TUESDAY	THREE	ELEVEN	SEABASE
WEDNESDAY	FOUR	TWELVE	INTERNATIONAL
THURSDAY	FIVE	A.M.	KENNEDY
FRIDAY	SIX	P.M.	-
SATURDAY	SEVEN	NOON	-

Table (1) Thirty Words Vocabulary for Experiment One

The reference phonemes used in the se experiments are a subset of the five vowels: $\ddot{a}$, $\bar{i}$, $\bar{e}$, $\ddot{u}$, and $\bar{u}$, and the four fricatives: n, s, f and t.

Ninety (three for each word in the vocabulary) words were used for formation of reference matrices. In the same way, ninety words were used as the test set. The utterances were spoken in a random order and at different times of day.

3.3.1 Variations of the Order of Segmentation

With the order of prediction and reference phonemes fixed at ten and CV1 (see table (2)) respectively, the order of segmentation was varied from five to forty. Figure (3) shows the number of errors and near errors under different orders of segmentation.

Reference Phoneme Set Label	Composition
C1	n, f, s, t
V1	ä, $\bar{i}$, $\bar{e}$, ü, $\bar{a}$
CV1	n, f, s, t, ä, $\bar{i}$, $\bar{e}$, ü
CV2	n, f, s, t, $\bar{i}$, $\bar{e}$, ü
CV3	n, f, s, t, ä, $\bar{i}$, $\bar{e}$, ü, $\bar{a}$

Table (2) Composition of Reference Phoneme Sets

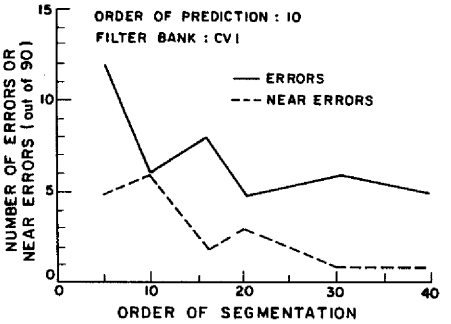

Figure (3) Variations of the Order of Segmentation: Error Rate, vs. Order of Segmentation

3.3.2 Variations of the Order of Prediction

With the order of segmentation and the reference phonemes fixed at twenty and CV1 respectively, the order of prediction was varied from six to eighteen. Figure (4) shows the number of errors and near errors under different orders of prediction.

3.3.3 Variations of Filter Bank

With the order of segmentation and the order of prediction fixed at twenty and ten respectively, the filter banks, i.e. the set of feature space phonemes were varied. Table (2) gives the composition of each filter bank. No plotting (such as that of figure (3) and (4)) is given because

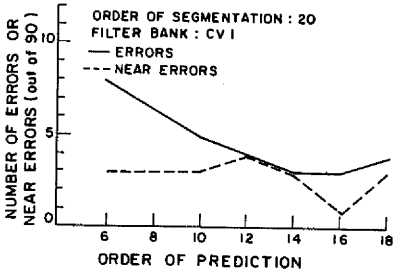

Figure (4) Variation of the Order of Prediction: Error Rate vs. Order of Prediction.

the independent variable is not orderable.

Table (3) shows that the use of the filter bank made of only four consonant filters (C1) resulted in the worst error rate; nevertheless, it was able to achieve a recognition rate of 72/90. This recognition rate shows that consonant filters are significant in a filter bank. It is found that at least nine filters, including both consonant and vowel filters, should be carefully chosen to achieve acceptable results.

3.4 The Second Experiment

In the first experiment, the optimal orders of prediction and segmentation were found to be around sixteen and twenty respectively. It was found also that about nine reference phonemes are sufficient for reasonable recognition accuracy, and that both consonant and vowel should be included in the set.

However, two questions were not answered by the first experiment: (1) what is the recognition power of the system? (2) what is the resolving power of the system in discriminating reference templates? The second experiment is designed to answer these two questions.

The experimental procedure is the same as in experiment one (section 3.3.). In this experiment the speaker was a different male with English as his second language. The vocabulary consisted of ten words only - one to ten. The test set was increased to 380 (38 for each word) to give

WORDS	FILTER BANK				
	C1	V1	CV1	CV2	CV3
NO	-	2/0*	-	-	-
MONDAY	-	-	-	0/1	-
FRIDAY	1/0	1/0	-	-	-
SATURDAY	0/1	1/0	-	-	-
SUNDAY	-	1/0	-	-	-
ONE	-	1/0	-	-	-
TWO	2/0	1/0	-	-	-
THREE	1/0	-	-	-	-
FOUR	2/0	-	-	1/2	-
FIVE	2/0	3/0	2/0	2/0	1/0
EIGHT	2/0	-	-	-	-
NINE	-	-	-	-	1/1
TEN	0/1	-	-	0/2	-
TWELVE	-	1/0	1/0	0/1	-
A.M.	1/1	-	0/1	1/0	1/0
P.M.	-	-	1/0	-	0/1
NOON	3/0	-	1/0	2/0	1/0
MIDNIGHT	-	1/0	0/1	-	1/0
ANY	2/0	-	-	-	-
EITHER	1/0	1/0	-	-	-
KENNEDY	1/0	-	-	-	-
TOTAL	18/3	13/0	5/2	6/6	5/2

Errors/Near-Errors, Blank = 0/0

Table (3) Variations of Filter Banks.

a more significant estimation of the recognition rate. The reference phonemes consisted of the ten phonemes : $\ddot{a}$,n,$\ddot{u}$, $\overline{e}$, $\overline{o}$, $\overline{i}$,f ,s,e, and $\overline{a}$.

In this experiment, 380 test utterances, spoken in a random order over several days, were sampled. Four words were classified incorrectly - once for the word ONE, twice for the word THREE and once for the word NINE.

Tables (4) and (5) give the mean, standard deviation, maximum, and mimimum of the Shortest Distance (SD) and Distance Difference (DD) of the experiment. The SD is the smallest distance value obtained when an unknown template is matched with the reference templates. The DD is the difference between the SD and the second shortest distance. In general, the lower the SD or the higher the DD, the better is the resolving power. The statistics were collected only

from the correctly recognized words.

From table (4), the worst (largest) SD is for the word THREE, which agrees with the experimental result, for there are two recognition errors for this word. The best (shortest) distance is for the word SIX.

	MEAN	STANDARD DEVIATION	MAXIMUM	MINIMUM
ONE	.183	.028	.258	.135
TWO	.166	.019	.195	.128
THREE	.265	.029	.350	.218
FOUR	.124	.031	.244	.085
FIVE	.224	.045	.301	.146
SIX	.103	.012	.142	.081
SEVEN	.171	.020	.223	.143
EIGHT	.179	.029	.228	.113
NINE	.220	.037	.292	.154
TEN	.182	.031	.244	.132

Table (4) Statistics of the Shortest Distance (SD)

	MEAN	STANDARD DEVIATION	MAXIMUM	MINIMUM
ONE	.174	.048	.247	.034
TWO	.149	.040	.240	.062
THREE	.147	.056	.250	.018
FOUR	.192	.042	.257	.083
FIVE	.092	.032	.148	.008
SIX	.198	.018	.230	.148
SEVEN	.138	.026	.194	.084
EIGHT	.181	.024	.229	.127
NINE	.108	.045	.201	.017
TEN	.167	.030	.217	.088

Table (5) Statistics of the Distance Difference (DD)

From table (5), the words that contain errors - ONE, THREE and NINE - have the largest deviation. This observation is consistent with the meaning of the standard deviation. The wider the spread of the DD, the easier it is to have errors. The words, SIX and EIGHT have the largest mean DD and the smallest standard deviation; therefore, they can be considered as the best recognizable words.

4. Hardware Implementation

Techniques discussed earlier in this paper are particularly suitable to direct hardware implementation. Figure (5) shows one such implementation. The speech signal is fed in parallel to a number of prediction filters. Each filter is tuned to predict one of the sounds of language and the difference between the predicted and the actual value appears in the output of the filter. Such bank of filters correspond to the set of the reference phonemes in the equivalente software implementation. Design of such filters is straight forward and well studied. Their regular structure resembles that of systolic hardware and therefore suitable to VLSI implementation.

Such filters can also be implemented, rather easily, using the digital signal processors. Depending on the word length, order of prediction, and the speed and power of the signal processor, one processor may be able to implement one or more filters with the possibility of implementing all filters using a single signal processor.

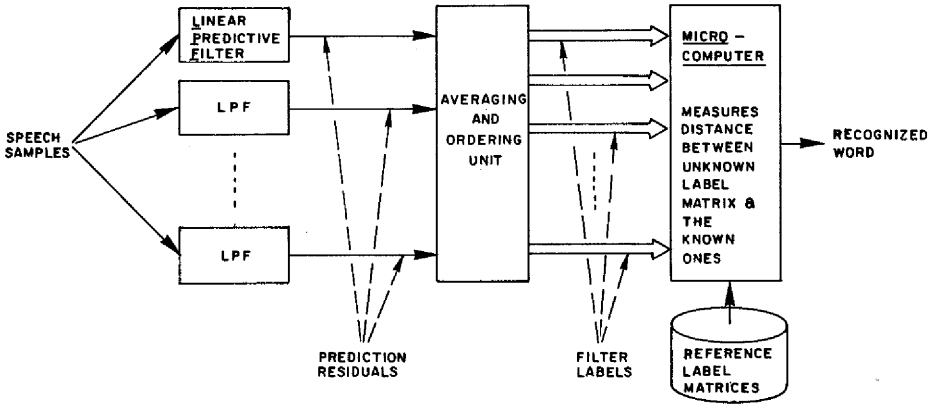

Fig (5): Hardware Preprocessor

The "averaging and ordering unit" integrates the square of each filter output and displays at its outputs the codes corresponding to the filter labels arranged in the order of the integration values. Integration can be performed using simple analogue techniques and the ordering can be performed using very simple single chip microcomputers. Slow rate of label vector updating (around twenty per word) permits the use of very simple and cost effective techniques in its design

and implementation. Measuring similarity between the label matrices must be performed using conventional equipment.

5. Summary and Conclusions

It was shown that similarity measurements can also be performed in the label space. Being a redundant space, techniques of removing some of these redundancies were discussed.

It was further shown that the distance measure in the space of LP feature is costly and provides more information than what may be required for certain type of questions.

Finally ways of combining the two observations in the label space and the space of the LP features were discussed. It was demonstrated that it can lead to a sufficiently powerful framework for word recognition systems, with computationally less complex and less expensive calculations.

Some of the important reasons for using the proposed system are as follows:

1- The volume of needed programs is small and can be easily implemented in very small systems.

2- Processing can be very fast. There are very good techniques for its further reduction, depending on the number of words in the lexicon versus the number of reference sounds.

3- System is easily amendable to hardware implementation, using existing modules in the market and/or custom VLSI implementation.

4- With little modification the same technique can be applied to the initial phonological labelling phase of the continuous speech recognition.

6. Acknowledgements

I would like to thank Dr. Paolo Franchi of IBM Italia, who motivated my initial interest in this area. Thanks are also due to IBM World Trade Corp., University of Waterloo, and the

National Science and Engineering Research Council of Canada who have financially supported this project at its different stages of development.

References

[Atal1] - Atal, B.S., M.R. Schroeder, "Adaptive Predictive Coding of Speech Signals", Bell System Tech. Journal, 49, 1970.

[Atal2] - Atal, B.S., S.L. Hanauer, "Speech Analysis and Synthesis by Linear Prediction of the Speech Wave", The Journal of the Acoustical Society of America, Vol. 50, No. 2, 1971.

[Broad] - Broad, D.J., J.E. Shoup, "Concepts for Acoustic Phonetic Recognition", Speech Recognition, (Ed. D.R. Reddy), Academic Press, 1975.

[Coker] - Coker, M.J., S.F. Boll, "An Improved Isolation Word Recognition System Based Upon the Linear Prediction Residual", pp. 206-209.

[Flannagan] - Flannagan, J.L., "Speech Analysis Synthesis and Perception", Springer Verlag, 1972.

[Gupta] - Gupta, V.N., J.K. Bryan, and J.N. Gowdy, "A Speaker-Independent Speech Recognition System Based on Linear Prediction", IEEE trans.. on Vol. ASSP 26, No. 1, Feb. 1978.

[Itakura] - Itakura, F., "Minimum Prediction Residual Principle Applied to Speech Recognition", IEEE Trans. on ASSP, Feb. 1975.

[Jelinek] - Jelinek, F., "Continuous Speech Recognition by Statistical Methods", Proceedings of the IEEE, Vol. 64, No. 4, April 1976.

[Makhoul] - Makhoul, J., "Linear Prediction: A Tutorial Review", Proceedings of IEEE, Vol. 63, No. 4, April 1975.

[Reddy] - Reddy, D.R., "Speech Recognition by Machine: A Review", Proceedings of

IEEE, Vol. 64, No. 4, April 1976.

[Sakoe] - Sakoe, H., S. Chiba, "Dynamic Programming Algorithm Optimization for Spoken Word Recognition", IEEE Trans. on ASSP, Vol. 26, No. 1, Feb. 1978.

[Fujiskai] - Fujiskai, H., Y. Sato, "Evaluation and Comparison of Features in Speech Recognition", Faculty of Eng., Univ. of Tokyo, Eng. Res. Inst., Vol. 32, pp. 213-218, 1973.

[Shoup] - Shoup, J.E., "Phoneme Selection for Studies in Automatic Speech Recognition", The Journal of the Acoustical Society of America, Vol. 34, No. 4, April 1962.

[Souza] - Souza, P.V., "Statistical Tests and Distance Measures for LPC Coefficients", IEEE Trans. on ASSP, Vol. 25, No. 6, Dec. 1977.

[Rabiner] - Rabiner, L.R., "On Creating Reference Templates for Speaker Independent Recognition of Isolated Words", IEEE Trans. on ASSP, 26, No. 1, Feb. 1978.

[White] - White, G.M., R.B. Neely, "Speech Recognition Experiments with Linear Prediction, Bandpass Filtering, and Dynamic Programming", IEEE Trans. on ASSP, Vol. 24, No. 2, April 1976.

[Tribolet] - Tribolet, J.M., L.R. Rabiner, M.M. Sondhi, "Statistical Properties of an LPC Distance Measure", IEEE Trans. ASSP, Vol. 27, No. 5, Oct. 1979.

[Levinson] - Levinson, S.E., "Interactive Clustering Techniques for Selecting Speaker. Independent Reference Templates for Isolated Word Recognition", IEEE Trans. ASSP, Vol. 27, No. 2, 1979.