CMU-CS-81-134

(12)

Effect of Reference Set Selection on Speaker Dependent Speech Recognition

Frame Compression in Isolated Word Recognition

Zongge Li, Fil Alleva and Raj Reddy

23 July 1981

DEPARTMENT
of
COMPUTER SCIENCE

Carnegie-Mellon University

14 CMU-CS-81-134

6

Effect of Reference Set Selection on Speaker Dependent Speech Recognition.

(Frame Compression in Isolated Word Recognition)

10 Zongge Li, Fil Alleva [redacted] Raj Reddy

11 23 Jul[redacted]981

12 18

Carnegie-Mellon University
Computer Science Department

15 F33615-78-C-1551, ARPA Order-3597

This research was sponsored in part by the National Science Foundation, Grant MCS-7825824 and in part by the Defense Advanced Research Projects Agency (DOD), ARPA Order No. 3597, monitored by the Air Force Avionics Laboratory Under Contract F33615-78-C-1551.

The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

403081

Table of Contents

1. Abstract

This paper describes an algorithm for compressing the spectral representation of an utterance along the time axis while keeplng the main features intact. The goal of the algorithm is to save template storage space and to reduce the time requlred for recognition. For 8 speakers, 5 data sets each, the results indicated that we can save about 40% of the template space and 35% of the recognition time with only a slightly higher error rate.

2. Introduction

In speech recognition dynamic programming is commonly used to time align the test utterance and reference utterance frame by frame.[1][2]

In checking the feature parameters (spectral data In our case) in one utterance, we often find contiguous frames which have almost the same feature parameters. We can say they are similar to each other and need not all to be matched one by one. We can just keep one frame and delete others. This process is called "frame compression" because several frames are compressed into one frame.

It Is obvlous that frame compression would save space and time for warping. It is also possible that frame compression keeps the main features of the utterance when it is done appropriately.

From thls Idea, we developed an algorithm for frame compression and tested it on a large speech data set. The results indicate the feasibility of this approach.

3. The Algorithm

Flg. 3-1 shows the flow chart of the compression procedure. The Input is the uncompressed spectral data (15 coefficients for every frame and 4 bits per coefficient). Let us assume that an utterance has N frames labeled 0 to N-1. The compressing process consists of calculating Euclidean distances between a frame and a number of its neighbors, then marking the frame for either retention or deletion. In the figure, d[-1] is the distance between frame i and frame i-1. d[+j] is the distance between frame i and frame i+j (j=1,2,3). s[i] is the mark which indicates whether frame i should be deleted (with "-" mark) or not. The output then consists of the frames with a "+" mark only. The decision of the mark depends on the distances compared with a threshold T. (At present a value of T=25 is used). Table 3-1 shows a typical segmentation trace of the utterance "B".

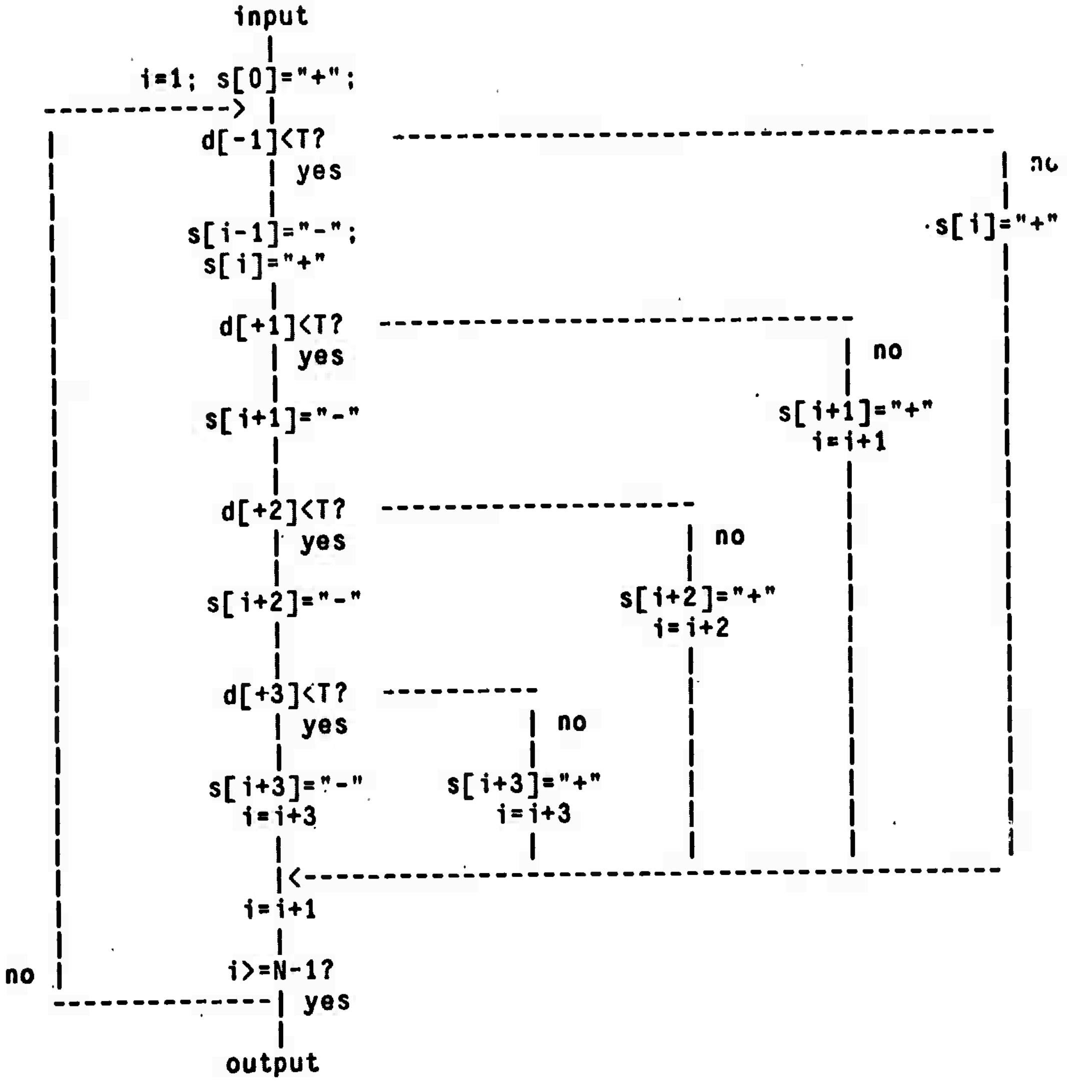

Figure 3-1: The flow chart of the frame compression algorithm.

Table 3-1: As an illustration of the operation of the algorithm as shown in the flow chart, this table gives the coefficients, distances and mark for every frame. In the table, c0 to c14 are the 15 coefficients. d[-1], d[+1], d[+2] and d[+3] are the four distances. fn is the frame number and s is the mark which indicates that the corresponding frame should be deleted ("-") or not (" + ").

c0	c1	c2	c3	c4	c5	c6	c7	c8	c9	c10	c11	c12	c13	c14	d[-1]	d[+1]	d[+2]	d[+3]	fn	s
1	-1	0	0	0	0	0	1	0	1	0	0	0	1	-2	0	0	0	0	0	+
5	-5	-3	-3	-2	-1	0	4	0	1	0	1	0	1	-2	65	0	0	0	1	+
6	-1	-3	-3	-1	-1	-1	-1	3	4	3	0	0	-1	-4	80	0	0	0	2	+
7	4	-5	-2	-5	0	-1	0	2	4	5	-1	-3	-1	-3	65	0	0	0	3	-
7	5	-6	-3	-6	2	0	0	0	4	6	-1	-4	0	-4	17	18	23	26	4	+
7	4	-5	-4	-4	0	0	0	0	3	6	1	-4	1	-3	0	0	0	0	5	-
7	4	-5	-5	-6	2	0	0	1	1	7	0	-2	1	-4	0	0	0	0	6	-
7	4	-5	-4	-6	2	0	0	1	1	7	0	-1	-1	-3	0	0	0	0	7	-
7	4	-5	-5	-5	1	0	-1	2	0	6	1	-2	0	-3	10	11	14	16	8	+
7	4	-5	-4	-7	2	0	0	1	0	7	0	-1	0	-3	0	0	0	0	9	-
7	4	-5	-5	-6	2	0	0	1	0	7	-1	0	0	-4	0	0	0	0	10	-
7	4	-6	-4	-6	2	0	0	1	0	7	-1	0	0	-2	0	0	0	0	11	-
7	4	-6	-4	-7	2	0	0	0	0	7	-1	0	0	-4	6	9	14	12	12	+
7	4	-6	-4	-6	1	0	0	1	1	6	0	1	-1	-3	0	0	0	0	13	-
7	4	-6	-4	-5	1	0	-1	2	0	7	0	1	-1	-3	0	0	0	0	14	-
7	4	-6	-4	-5	1	0	0	1	0	7	0	1	-2	-4	0	0	0	0	15	-
7	4	-6	-5	-3	0	-1	-1	1	2	7	-2	0	0	-4	21	12	16	12	16	+
7	4	-6	-5	-4	0	0	0	0	0	7	-1	1	-1	-3	0	0	0	0	17	-
7	4	-6	-6	-3	0	-1	0	0	1	7	0	2	-2	-4	0	0	0	0	18	-
7	4	-7	-5	-3	0	0	0	0	1	7	-1	2	-1	-3	0	0	0	0	19	-
7	4	-7	-5	-4	0	0	0	1	1	7	0	1	-1	-4	5	8	6	9	20	+
7	4	-7	-6	-3	0	0	0	0	0	7	-1	2	0	-3	0	0	0	0	21	-
7	4	-7	-4	-3	-1	-1	-1	1	2	7	0	1	-1	-4	0	0	0	0	22	-
7	4	-7	-4	-3	-1	-1	-1	1	1	6	-1	2	0	-4	0	0	0	0	23	-
7	4	-7	-4	-3	-1	-1	-1	1	1	6	1	2	0	-5	5	6	8	5	24	+
7	4	-7	-4	-2	-1	-1	0	1	0	7	1	1	0	-4	0	0	0	0	25	-
7	4	-7	-5	-2	-2	-2	0	1	0	6	1	1	0	-4	0	0	0	0	26	-
7	4	-7	-4	-2	-2	-1	0	1	1	5	0	2	0	-5	0	0	0	0	27	-
7	4	-7	-5	-1	-1	-1	0	0	1	6	1	2	0	-5	6	13	8	22	28	+
7	4	-7	-4	-1	-1	-2	-2	1	2	6	0	2	0	-7	0	0	0	0	29	-
7	4	-7	-5	-1	-1	-1	-1	0	2	7	0	2	0	-7	0	0	0	0	30	-
7	4	-7	-6	-1	-1	-1	-3	2	2	7	0	2	-1	-7	0	0	0	0	31	-
7	4	-7	-6	0	-2	-2	0	0	2	5	2	3	-1	-6	26	0	0	0	32	-
7	3	-7	-7	0	-1	-1	-2	0	2	6	3	2	0	-5	13	4	5	18	33	+
7	3	-7	-7	1	-1	-1	-2	0	1	7	3	1	0	-5	0	0	0	0	34	-
7	3	-7	-7	0	-1	0	-3	0	1	7	3	2	0	-4	0	0	0	0	35	-
7	2	-7	-4	0	-2	-1	-4	1	2	7	3	2	0	-4	0	0	0	0	36	-
7	2	-7	-2	-1	-1	-4	0	0	2	7	3	1	0	-3	34	0	0	0	37	+
7	0	-7	-1	-3	1	-1	-4	0	1	7	4	1	0	-3	40	0	0	0	38	+
7	-4	-7	-2	2	-4	0	0	-2	0	5	5	0	0	-2	96	0	0	0	39	+
7	-3	-7	-1	1	0	-2	-2	0	0	4	5	0	-2	-2	36	0	0	0	40	-
7	-3	-7	0	-1	0	-1	-1	0	0	4	4	1	-1	-4	14	32	0	0	41	+
7	-4	-7	-3	0	-1	2	-3	0	0	3	4	-1	0	-3	0	0	0	0	42	+

4. Results of Experiment and Discussion

The experiment was performed on a VAX-11/780 computer using the Cicada2 system as described elsewhere.[3][4] The experiment was done by using the data of 4 male speakers and 4 female speakers. For every speaker five data sets were used as test sets and one data set was used as reference sεt[5]. Each set consists of 36 utterances (10 digits and the 26 letters of the alphabet). All utterances have automatically determined endpoints. Table 4-1 gives the recognition results.

From Table 4-1 we can see that accuracy of using compressed data is somewhat inferior to that using noncompressed data. The overall error rate (in percent) is calculated by sum/total number of test utterances (= 1440).

Table 4-1: Comparison of compression vs noncompression

speaker	errors(com)	errors(noncom)
ds	26	21
fa	14	9
gg	23	24
jl	14	27
ma	33	22
ms	19	13
rp	4	6
sw	33	34
sum	166	156
%	11.5	10.8

Table 4-2 shows the percentage of the frames deleted from an utterance for four speakers. On the average about 40% frames were deleted. This indicates that we can save about 40% template space and about 35% warping time (the saving is less because of the extra computation time needed for compression).

Table 4·2: Data Reduction In Percent

speaker	percent
ds	45.6
fa	33.4
gg	39.4
sw	43.6
average	40.5

5. Acknowledgement

The authors would like to express their gratitude to Dr. A. Rudnicky and Mr. A. Waibel for their patient reading and careful correction and to Mr. F. Alleva for his help in doing the experiment.

6. References

[1] Itakura, F.
Minimum prediction residual applied to speech recognition
IEEE Transactions on Acoustics, Speech and Signal Processing
23(1):67-72, February,1975

[2] Sakoe,H. and Chiba,S.
A Dynamic Programming Approach to Continuous Speech Recognition
Proceedings of International Congress on Acoustics
Budapest,Hungary, Paper 20C-13, 1971.

[3] Waibel,A. and Yegnanarayana,B.
Comparative Study of Nonlinear Time Warping Techniques in
Isolated Word Speech Recognition Systems
Technical Roport of Carnegie-Mellon University, 1981

[4] Waibel,A., Krishnan,N. and Reddy,R.
Minimizing Computational Cost for Dynamic Programming Algorithms
Technical Report of Carnegie-Mellon University, 1981

[5] Li,Z., Alleva,F., and Reddy,R.
The effect of reference set selection on speaker dependent
speech recognition
Technical Report of Carnegie-Mellon University, 1981

Frame Compression in Isolated Word Recognition

Z. Li
Department of Computer Science
Carnegie-Mellon University
and
Fudan University, Shanghai, China

R. Reddy
Department of Computer Science
Carnegie-Mellon University

June 20, 1981

This research was sponsored in part by the National Science Foundation, Grant MCS 7825824 and in part by the Defense Advanced Research Projects Agency (DOD), ARPA Order No 3597, monitored by the Air Force Avionics Laboratory Under Contract F33615-78-C-1551. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Defense Advanced Research Projects Agency or the US Government.

Table of Contents

Abstract

In both speaker dependent and independent word recognition the selection of the reference templates is recognized as a crucial step in regards to the final accuracy of the system. Presented here for a speaker dependent system, is an algorithm which chooses a reference template for each word in the vocabulary from a set of N exemplars. The goal of the algorithm is to produce a reference set that minimizes the worst matching behavior and total error over the N sets of exemplars. The results of the experiments presented here show a reduction in the average error rate from 16.4% to 10.2% over a set of 4 male talkers and 4 female talkers.

1. Introduction

An Important problem in isolated word recognition is the creation and or selection of the reference templates. Techniques for clustering of templates [3] [4] have been developed which yield multiple reference patterns in speaker independent systems. Our experiments Indicate that the selection of the reference templates in the speaker dependent case has a significant effect on the recognition accuracy obtained. The technique presented in this paper selects a single optimal template for each vocabulary item based on the internal consistency of matches in an initial training set. The results we obtained with our template selection algorithm produce recognition results superior in all cases to those results obtained when no template selection is done.

2. Word Recognition System

Figure 2-1 shows a flow diagram of the system [1] used in these experiments. The speech data used in the experiments consists of 10 repetitions of the alphabet and digits (36 utterances) by 8 talkers (4 male, 4 female). Each talker completed two repetitions a day over period of five days. Each repetition was spoken in a different a randomized order. The recording was done in an office environment using a noise canceling microphone and high quality tape recorder. The recorded speech was then low pass filtered at 4.5 kHz and digitized at 10 kHz.

2.1. Signal Processing

The raw digitized samples are taken as the input to a 256 pt. discrete Fourier analysis, using a 20.0msec. window stepped at 10.0msec. intervals. The results of the Fourier analysis are then reduced to 16 coefficients by summing adjacent values in the spectrum according to the mel scale (see table 2-1). These 16 coefficients are then converted to log dB. Begin-End analysis proceeds on the log dB signal by computing for each frame,[2] the average energy and the difference between high

[2] Frames are defined as a set of 16 coefficients that represent 20.0msec of signal.

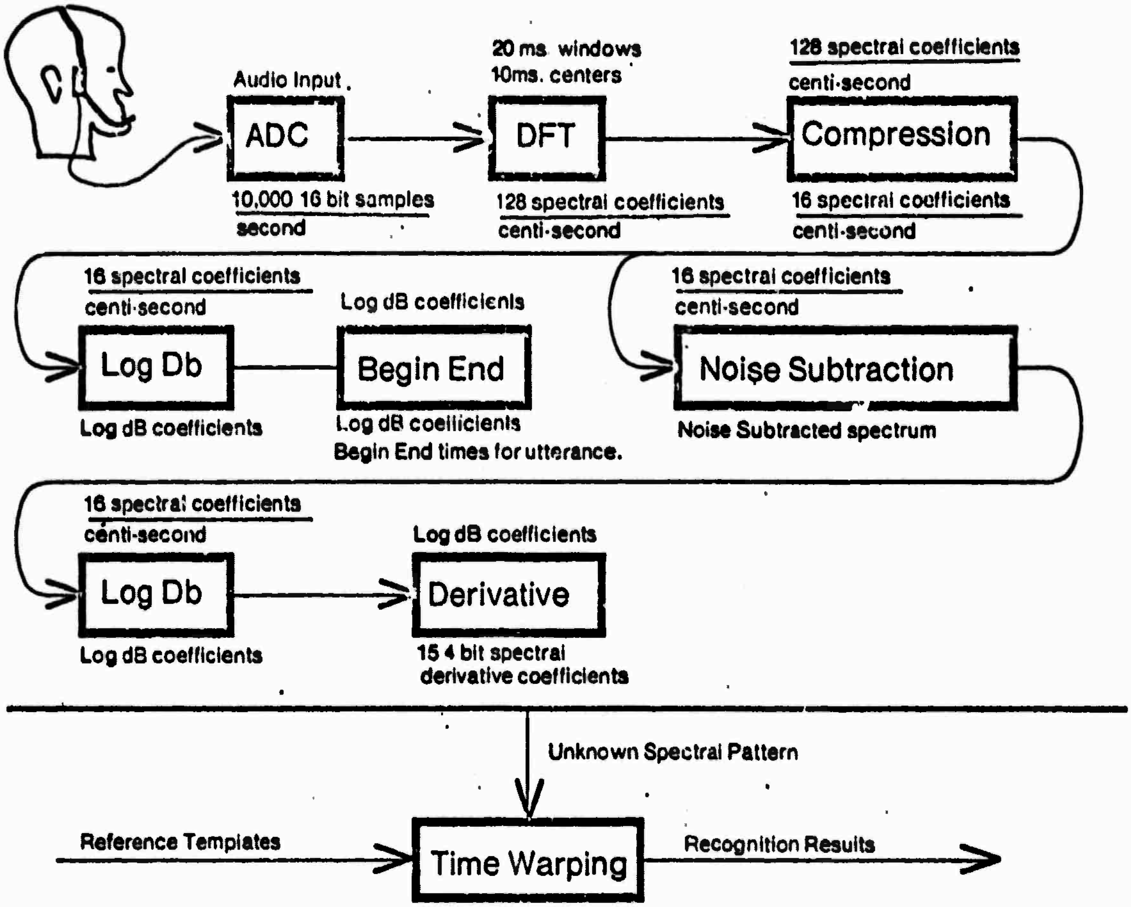
Audio Input.
ADC
10,000 16 bit samples / second
20 ms windows
10ms. centers
DFT
128 spectral coefficients / centi-second
128 spectral coefficients / centi-second
Compression
16 spectral coefficients / centi-second
16 spectral coefficients / centi-second
Log Db
Log dB coefficients
Log dB coefficients
Begin End
Log dB coefficients
Begin End times for utterance.
16 spectral coefficients / centi-second
Noise Subtraction
Noise Subtracted spectrum
16 spectral coefficients / centi-second
Log Db
Log dB coefficients
Log dB coefficients
Derivative
15 4 bit spectral derivative coefficients
Unknown Spectral Pattern
Reference Templates
Time Warping
Recognition Results

Figure 2.1: Flow Diagram of System

and low frequency energy content, these two parameters are then used in the begin-end analysis.

Noise subtraction is accomplished by computing an average noise spectrum and subtracting it from each frame of the signal. If the energy level of a coefficient is below the average energy per coefficient in the noise spectrum after the noise spectrum is subtracted then that coefficient is set to that average energy level. Finally the coefficients are reduced to a 4 bit magnitude by taking the derivative with respect to frequency.

2.2. Warping

The dynamic programing method used is the Itakura warping technique [2]. Although there are several other dynamic time warping algorithms which have been proposed, the Itakura warping appears to give the most consistent results over a variety of conditions. The metric used to measure the difference between the test and reference is a euclidean distance.

Filter	Range of DFT Samples	Frequency Range
0	0 - 2	0 - 98 Hz.
1	2 - 6	98 - 254 Hz.
2	6 - 10	254 - 410 Hz.
3	10 - 14	410 - 566 Hz.
4	14 - 18	566 - 722 Hz.
5	18 - 22	722 - 878 Hz.
6	22 - 26	878 - 1034 Hz.
7	26 - 30	1034 - 1191 Hz.
8	30 - 35	1191 - 1387 Hz.
9	35 - 41	1387 - 1622 Hz.
10	41 - 48	1622 - 1896 Hz.
11	48 - 57	1896 - 2248 Hz.
12	57 - 68	2248 - 2677 Hz.
13	68 - 81	2677 - 3185 Hz.
14	81 - 97	3185 - 3809 Hz.
15	97 - 116	3809 - 4551 Hz.

Table 2-1: Mel Scale Frequency Boundaries[1]

3. Reference Template Selection

As previously stated the goal of the template selection algorithm is to chose a reference template set from the training set that will provide the best match to the training set. For the purposes of our discussion the first 5 repetitions of each speaker in our data base will be designated as the training data sets. The last 5 repetitions will be designated as the test data sets. Initially we are interested in what the results of the recognition are if we do no template selection and simply allow each of the training data sets to serve in turn as the reference templates for the test data sets. These results are presented table 3-1.

As can be seen, the error rate varies a great deal, depending on which data set is used for the reference templates. When template selection is done, we will take advantage of the variance in pronunciation and build a composite set of reference templates that exhibits a matching behavior better than any one of the original training sets.

3.1. Selection Algorithm

The algorithm proceeds by addressing the problem of templates belonging to utterances that are easily confused. The key point being that the differences between these templates is not always large enough to discriminate them correctly when matched with an unknown utterance. By carefully selecting templates from the training sets we can increase the difference between confusable

[1] The range of DFT samples included in each filter is determined by the size of the DFT (256 points in this case) and the range of frequencies present in the signal (0 - 5000 Hz. in this case). The end samples of each filter are given half their weight in the filter which is composed of the sum the specified DFT samples.

Male Speaker Reference	M1 Error Rate	M2 Error Rate	M3 Error Rate	M4 Error Rate
1	17.2%	22.8%	32.8%	7.2%
2	11.1%	12.2%	23.3%	9.4%
3	7.9%	12.8%	27.8%	5.0%
4	7.9%	8.3%	22.8%	12.2%
5	7.2%	11.7%	24.4%	5.0%
Average	10.2%	13.6%	26.2%	7.8%
Best	7.2%	8.3%	22.8%	5.0%

Female Speaker Reference	F1 Error Rate	F2 Error Rate	F3 Error Rate	F4 Error Rate
1	10.0%	21.7%	15.0%	21.7%
2	17.2%	23.3%	17.8%	22.2%
3	15.7%	21.1%	16.1%	20.0%
4	13.9%	19.4%	16.7%	16.1%
5	12.2%	23.9%	17.8%	25.0%
Average	13.8%	21.9%	16.7%	21.0%
Best	10.0%	19.4%	15.0%	16.1%

Grand Average = 16.4%

Average of Best error rates = 13.1%

Table 3-1: Recognition results with no template selection.

templates thereby reducing the error rate otherwise obtained. In order to facilitate the discussion of the algorithm we shall designate $U[e,w]$ as the utterances in the training set, $M1[r,t,w]$ as the first choice matching behavior and $M2[r,t,w]$ as the second choice matching behavior, where

e = exemplar number, $e = 1,2,\ldots,N$
w = word number, $w = 1,2,\ldots,W$
r = reference exemplar, $r = 1,2,\ldots,N$
t = test exemplar, $t = 1,2,\ldots,N$

3.1.1. Candidate Selection

Consider the first choice matching behavior for word w' which we denote as $M1w'[r,t]$. In figure 3-1 we see an example of the first choice matching behavior for the vocabulary item "f". The first choice matching behavior for a particular word in a particular test dataset is defined as the score obtained and the word recognized given a particular reference dataset. In our example we see, for instance, that the "f" in test dataset *2* is indeed recognized as an "f" with a score of *53* when dataset *1* is used as the reference.

For each reference dataset we observe that there will be a range of scores obtained over the N test datasets. For each reference the worst score over the N test datasets is picked out and defined as the worst matching behavior for that reference. Let $WM1w'[r]$ be $Max\ M1w'[r,t]$ denote the worst

M1w'(rt)

		reference dataset				
		1	2	3	4	5
test dataset	1	0	50/x	54/f	44/f	64/f
	2	53/f	0	$\boxed{57/f}$	$\boxed{63/f}$	$\boxed{69/f}$
	3	55/f	53/f	0	46/f	53/f
	4	43/f	54/f	43/f	0	60/f
	5	$\boxed{62/f}$	$\boxed{60/f}$	52/f	52/f	0

WM1w'(r)	62	60	57	63	69

Figure 3-1: First Choice Matching Behavior of "f"

matching behavior for each reference *r* over the *t* test exemplars for word *w'*. In figure 3-1 the worst matching behavior for each of the *N* references is boxed.

Once the vector (*WM1w'[r]*) containing the worst matching behavior for each of the references is formed we choose the candidate template for *w'* as *U[r',w']* such that *r'* is the *Min WM1w'[r]* over the *N* references. That is. the reference dataset that has the minimum worst matching behavior becomes our candidate dataset. In our example the candidate template for *w'* is in dataset *3*.

3.1.2. Verification

In order to verify that *U[r',w']* is indeed the best candidate for *w'* we must establish that the matching behavior, *M1r'w'[t]*, over the *t* test exemplars does the following:

- Provides a correctly recognized word.
- Has a match distance that is less than any wrong first choice recognition.
- Has a match distance that is less than all second choice recognitions in *M2w'[r,t]* over all *r* for *r'*.

Using figure 3-1 we can check the first two conditions. We observe that dataset *3* meets the first condition since it provides a correct recognition of "f" for the other four datasets.

Checking the second condition we see that the "f" from *dataset 1* is recognized as an "x" with a score of 50 when *dataset 2* is used as the reference. This fails to meet the second condition since the recognition for "f" in our candidate dataset (*3*) has a score of 54. Since this is the case, choosing the "f" from *dataset 3* may possibly lead to inherent error in our selected dataset. This inherent error would arise if the "x" from *dataset 2* was chosen as part of our selected dataset. In that case an incorrect recognition result for the "f" from *dataset 1* would occur when the selected dataset was used as the reference.

Using figure 3-2 we can check the 'inal condition. We observe that the second choice matching behavior for "f" from *dataset 2* produces a score of 55 for an "s" from *dataset 2*. This can lead to inherent error in the same way as described for the second condition. Thus, the candidate template falls to meet the third condition.

M2w'(rt)

reference dataset

		1	2	3	4	5
test	1	67/x	70/f	54/f	72/s	56/s
dataset	2	78/m.	55/s	57/f	80/x	67/m
	3	62/m	46/s	0/f	79/x	60/m
	4	91/l	75/x	43/f	67/s	58/s
	5	96/f	99/s	52/f	80/s	87/x

First.Choice Matching
Behavior of dataset 3
M1w'(3,t)

Figure 3-2: Second Choice Matching Behavior of "f"

In the event that all of these conditions are satisfied then *U[r',w']* is a *good* template for *w'*, meaning that using it will not lead to inherent error when the selected dataset is used as the reference for our training datasets. However, for a majority of utterances a *good* template is not available since the discriminability between these utterances is too small. In order to minimize the inherent error, the choice of a *best w'* is made with reference to the entire set of training templates.

This procedure consists of selecting *p* additional candidates for *w'*. These candidates are chosen by increasing magnitude of *WM1w'[r]*. When one or more candidates have been selected for all *W* words, the inherent error for all combinations of the *p* candidates is computed among those words which did not have a *good* template. The combination of *W* templates that produces the least inherent error is then used as the selected template set. A potential draw back of this procedure is that *p* must be kept small since the number of combinations to compute grows exponentially with *p*. The data reported in this paper are based on template selection using a *p* of 2..

4. Recognition Results and Discussion

Speaker	New Error Rate	Average		Best	
		Error Rate	%Improvement	Error Rate	%Improvement
M1	5.6%	10.2%	45.0%	7.2%	22.2%
M2	7.8%	13.6%	42.6%	8.3%	6.0%
M3	18.3%	26.2%	30.1%	22.8%	19.7%
M4	1.1%	7.8%	85.8%	5.0%	78.0%
F1	7.2%	13.8%	47.8%	10.0%	28.0%
F2	16.7%	21.9%	23.7%	19.4%	13.9%
F3	10.6%	16.7%	36.5%	15.0%	29.3%
F4	14.4%	21.0%	31.4%	16.1%	10.5%
Average	10.2%	16.4%	42.8%	13.1%	25.9%

Table 4-1: Recognition Results using Template Selection

If we examine the results obtained (Table 4-1) when this algorithm for reference template selection is used, we see an improvement over the best results obtained for each speaker in the case where no template selection is done. The average expected improvement over the average expected recognition results is given as 42.8%. However this percentage might be expected to decrease with a smaller number of exemplars in the training set. Likewise a larger number of exemplars would probably result in a case of diminishing returns on recognition improvement. While this algorithm features the intuitively attractive feature of using a real template as opposed to a synthetic one, this feature will probably lead to poor results in the case of speaker-independent recognition.

References

[1] Alleva, F.A.
Cicada Users Manual.
Avalable from author at Carnegie-Mellon.

[2] Itakura, F.
Minimum Prediction Residual Applied to Speech Recognition.
IEEE Transactions on Acoustics, Speech and Signal Processing 23(1):67-72, February, 1975.

[3] Rabiner, L.R.
On Creating Reference Templates for Speaker Independent Recogniton of Isolated Words.
IEEE Transactions on Acoustics, Speech and Signal Processing 26(1):34-42, February, 1978.

[4] Rabiner, L.R., Levison, S.E., Rosenberg, A.E., and Wilpon, J.G.
Speaker Independent Recognition of Isolated Words.
IEEE Transactions on Acoustics, Speech and Signal Processing 27(4):336-349, August, 1979.