Institute of Phonetic Sciences,
University of Amsterdam,
Proceedings 20 (1996), 97-110.

ADAPTIVE VOWEL NORMALIZATION AND THE TIMIT ACOUSTIC PHONETIC SPEECH CORPUS

David Weenink

Abstract

In this paper the adaptive model for speaker normalization as presented in Weenink & Pols (1993) is tested with material from the TIMIT speech corpus. Normalization was performed in that model by adapting the biases in a neural net. The neural net at that time was trained on formant frequency values from vowels excised from clearly articulated short speech utterances from men, women and children. We tested the model anew, now with principal components derived from a bandfilter analysis of vowel segments from the TIMIT corpus. These tests show that the model does not work equally well with this more variable input. Alternatives are proposed.

1. Introduction

In Weenink & Pols (1993) an adaptive model for vowel normalization is presented. The model presents vowel identification as an analogon of the learning and classification with a neural net of the feedforward type: vowel identification is considered to be based on the integration of a number of decisions. Each decision is of the same simple form: decide whenever accumulated evidence exceeds a certain threshold (bias). Changing this bias can influence a decision and consequently the identification. Geometrically one can visualize this bias change as the parallel movement of a decision plane between two classes. Moving the plane influences the potential class membership. The model was then verified with formant frequencies measured from the vowels of a small speech database with carefully pronounced short Dutch utterances by male and female speakers and children. It was conjectured that the model should work equally well with other frequency representations besides formant frequencies.

In this paper we will put this model to test with another frequency representation: bandfilter spectra and the principal components derived from these spectra. Such data can be derived more consistently and more easily than formant data. There is ample evidence that the first two or three principal components have a very high correlation with the first two or three formant frequency values (Pols et al., 1969; Bloothooft, 1985). We therefore expect the model to behave similarly for principal components as it did for formant frequency values.

Another expansion with respect to testing the model involved the type and the size of the speech database. Instead of the small database derived from carefully pronounced short Dutch utterances we will use (a selection from) a much larger and

much more variable American-English speech corpus, the TIMIT database (Lamel et al., 1986).

The outline of this paper will be as follows: section 2 will start with a short description of the TIMIT speech corpus and how we have made it accessible for visualization, selection and analysis. This will be followed, in section 3, by a description of the data selections we made and the measurements that were performed, notably bandfilter analysis followed by principal component analysis and classification by neural nets of the feedforward type. Next the tests with the model will be presented, followed by a discussion and future plans.

2. The TIMIT acoustic-phonetic continuous speech corpus

The TIMIT speech corpus resulted from the joint efforts of several American speech research sites. The text corpus design was done by the Massachusetts Institute of Technology (MIT), Stanford Research Institute and Texas Instruments (TI). The speech was recorded at TI, transcribed at MIT, and has been maintained, verified and prepared for CDROM production by the American National Institute of Standards and Technology (NIST). The CDROM was made available by the LDC. The TIMIT corpus contains a total of 6300 sentences, 10 sentences spoken by each of 630 speakers from 8 major dialect regions of the United States of America, 70% of the speakers were male, 30% were female. A speaker's dialect region was defined as that geographical area were he lived during his childhood years. Dialect number 8 (Army Brat) was assigned to people who had moved around a lot during their childhood and to whom no particular dialect could be attributed. Table 1 shows the dialect distribution of the speakers.

Table 1. TIMIT dialect distribution of speakers. The fist column contains the dialect number, followed by the geographical region. The third, fourth and fifth columns contain the number of male speakers, female speakers and the total number of recorded speakers in the corresponding dialect region, respectively. The last two columns contain the number of male and female speakers in the *training* part of the data base. The last row contains the column sums.

dr	Region	# male	#female	Total	#male	#female
1	New England	31	18	49	24	14
2	Northern	71	31	102	53	23
3	North Midland	79	23	102	56	20
4	South Midland	69	31	100	53	15
5	Southern	62	36	98	45	25
6	New York City	30	16	46	22	13
7	Western	74	26	100	59	18
8	Army Brat	22	11	33	14	8
		438	192	630	326	136

The 10 sentences produced by each speaker consisted of 2 SA-type, 5 SX-type and 3 SI-type sentences. Only two different SA-type sentences were designed, sa1 and sa2, and they were spoken by all 630 speakers. The sa1 sentence is "She had your dark suit in greasy wash water all year" and the sa2 sentence is "Don't ask me to carry an oily rag like that". There were 450 different phonetically compact SX-type sentences, and, given that each speaker produced 5 of these sentences, each SX-

sentences was reproduced by 7 speakers (630 · 5 / 450). An example of an SX-type sentence is sx217: "How permanent are their records?". Finally, each speaker reproduced three unique utterances out of 1890 different phonetically diverse SI-type sentences. An example of such an SI-sentence is si1027: "Even then, if she took one step forward he could catch her". Besides the speech sound recordings, the TIMIT disc contains documentation, a pronouncing dictionary and transcriptions of all 6300 recorded sentences at the sentence level, the word level and the phonetic level. Part of this documentation will be reproduced in condensed form below. The pronouncing dictionary lists all different words used in the speech database with their 'standard' phonetic transcription and stress markers. The TIMIT phonetic transcription uses ASCII-symbols. Consequently, in table 2 we show the translation between TIMIT symbols and International Phonetic Association (IPA) phonetic symbols for the vowels in the database.

The speech and associated data is organized on the CDROM according to the following simple file hierachy:

/timit/<USAGE>/<DIALECT>/<SEX><SPEAKER>/<SENTENCE>.<TYPE>

<USAGE>	:= *train* or *test*
<DIALECT>	:= one of *dr1*, *dr2*, *dr3*, *dr4*, *dr5* *dr6*, *dr7* or *dr8*
<SEX>	:= *m* or *f* (male or female)
<SPEAKER>	:= unique speaker identification (3 characters and a digit)
<SENTENCE>	:= *sa* or *si* or *sx* followed by a *sentence number*
<TYPE>	:= one of *wav*, *txt*, *wrd* or *phn*

All 6300 recorded sentences are in separate binary files with extension `.wav`. These binary files with audio data have a special format by which they can be recognized and subsequently read by a specialized computer program that does not need external knowledge. In the binary files audio data is preceded by information about the data according to a prescribed format, the so called SPHERE header which is 1024 bytes long. We will henceforth call these files "NIST sound files". Each NIST sound file always starts with a standard sequence of 16 characters by which they can be identified: "`NIST_1A\n\s\s\s1024\n`", where `\n` and `\s` stand for the ASCII newline and space character, respectively. The rest of the header contains, among other things, information about the number of audio channels, the number of samples per channel, the number of bytes per sample, the sampling rate and the type of data compression. This information is sufficient to read the speech data that follow the header.

Unfortunately the accompanying description label files, with the same file *name* but file *extensions* `.txt`, `.wrd` or `.phn`, do not contain any identification information. These text files describe the recording at an ever increasing level of detail. Each file contains one or more lines, each line has three items that are separated by spaces. Each line starts with two natural numbers followed by a string. These two (sample) numbers, with the second number always larger than the first, mark a sample interval. Files with extension `.txt` contain only one line with the text of the sentence in the string, files with the extension `.wrd` contain a line for each word in the sentence and files with the extension `.phn` contain one line for each phonetic symbol that occurs. A `.phn` file always starts and ends with the symbol "h#".

In order to be able to visualize the speech corpus and (part of) the associated data we decided to extend the general speech analysis computer program *praat* (Boersma

& Weenink, 1996) with the possibility to recognize and read the audio files and phonetic label files and make them accessible in the program. As was described above, the recognition of a NIST audio file is simple since it always starts with the same identification string. Reading the data also is straightforward since all information for the interpretation of the data is contained within the header in a fixed format. There were some minor complications which have to do with endianness of the data and with compression. Although the NIST audio files in the TIMIT database are not in a compressed format, we nevertheless incorporated *alaw*, *μlaw* and *embedded-shorten* decompression algorithms in our program. In this way we can read other databases as well that have audio files in NIST format such as the Dutch Polyphone Database (Den Os et al., 1995), the Groningen corpus (Sulter & Schutte) and the Translanguage English Database (Lamel et al., 1994).

All the audio files on the TIMIT CDROM have a sampling frequeny of 16 kHz and are quantized with 16 bits per sample, although the actual quantization of many files is not better than approximately 12 bits / sample.

Table 2. Translation table of TIMIT-symbols to IPA symbols for vowels only. The first column shows the TIMIT-symbol, the second column shows the corresponding IPA-symbol. The vowel sounds like the vowel in the word given in the third column. The fourth column shows the TIMIT labeling of the word in the preceeding column. The last column indicates with a "+" those 13 vowels that were actually analysed in the present study.

TIMIT	IPA	Sounds as in word...	TIMIT labeling of word	S
iy	i	b**ee**t	bcl b **iy** tcl t	+
ih	ɪ	b**i**t	bcl b **ih** tcl t	+
eh	ɛ	b**e**t	bcl b **eh** tcl t	+
ey	e	b**ai**t	bcl b **ey** tcl t	+
ae	æ	b**a**t	bcl b **ae** tcl t	+
aa	ɑ	b**o**tt	bcl b **aa** tcl t	+
aw	aʊ	b**ou**t	bcl b **aw** tcl t	
ay	aɪ	b**i**te	bcl b **ay** tcl t	
ah	ʌ	b**u**t	bcl b **ah** tcl t	+
ao	ɔ	b**ough**t	bcl b **ao** tcl t	+
oy	ɔɪ	b**oy**	bcl b **oy**	
ow	o	b**oa**t	bcl b **ow** tcl t	+
uh	ʊ	b**oo**k	bcl b **uh** kcl k	+
uw	u	b**oo**t	bcl b **uw** tcl t	+
ux	ü	t**oo**t	tcl t **ux** tcl t	+
er	ɝ	b**ir**d	bcl b **er** dcl d	+
ax	ə	**a**bout	**ax** bcl b aw tcl t	
ix	ɨ	d**e**bit	dcl d eh bcl b **ix** tcl t	
axr	ɚ	butt**er**	bcl b ah dx **axr**	
ax-h	ʌ	s**u**spect	s **ax-h** s pcl p eh kcl k tcl t	

Even though reading the accompanying text files is much simpler, a robust TIMIT label file recognizer can not be build because the label file format does not contain unique identifiable parts. The file contents itself is not sufficient either, since the numbers that mark the intervals are not given in seconds but are sample numbers and therefore depend on the sampling frequency. Despite these facts we incorporated in

the praat program a heuristic "TIMIT label file recognizer" for `.wrd` and `.phn` files based on the following algorithm:

IF there are at least two lines in the file AND
each line has three items, two numbers and one string, AND
number1 ≥ 0 AND number2 > number1 AND
number3 ≥ number2 AND number4 > number3 AND
((string1 equals "h#" AND string2 equals a phonetic label) OR
(string1 is lowercase AND string2 is lowercase))
THEN the file is a TIMIT label file.

To be able to select specific phonemes in the TIMIT speech corpus, such as, for example, vowels in stressed or unstressed syllables in penultimate word position, we decided to build a database with information about every phoneme in the corpus. This derived database is in the form of a simple text file with one line per phoneme. The are a total of 241,225 lines in the file. Each line contains the following fields:

phoneme string
phoneme type (Stop, Affricate, Fricative, Nasal, Glide&Semivowel, Vowel, Other)
start time of the phone in the sentence in seconds
end time (s)
duration (s) = end time - start time
stress value (0, 1, or 2)
number of syllables (≥1)
phoneme on the left
    type of the phoneme on the left
phoneme on the right
    type of the phoneme on the right
sentence identification
speaker identification
sex of the speaker (m or f)
dialect of the speaker (1..8)
usage type (train or test)

The *start time* and the *end time* were obtained by dividing the sample numbers that mark the phoneme by 16,000, the sampling frequency of the audio files. The fields *sentence identification* through *usage type* can be obtained from the directory name and the file name. Because the `.phn` label files do not contain any stress information we could only rely on the stress information in the pronouncing dictionary. However, the realized pronunciation often differs from the ideal pronunciation in the dictionary. This means that to find the position in the realized phoneme string where the stress occurs, we have to find a match between the *realized* phoneme string and the *ideal* phoneme string. The dynamic programming matching algorithm goes as follows:

for each `.phn` phonetic label file do
    read the realized phonene string
    construct the ideal phoneme string from the `.wrd` word label file
    construct the associated cost matrix
    find the optimal path through the matrix
    assign the realized stress
endfor

The ideal phoneme string for a sentence can be constructed from the word label file by concatenating all the phonemes from the word entries in the pronouncing

dictionary. Because in the phonetic label files the plosives have closure and burst separately labeled, we choose to adapt the dictionary by inserting before every plosive (b, d, g, p, t, k, jh and ch) the corresponding closure string. The realized phoneme string is simply the concatenation of all the labels in the phonetic label file. With these two strings we can construct a cost matrix. A matrix element at position [i][j] measures the cost of the confusion of realized phoneme [i] with ideal phoneme [j]. By varying the costs associated with a particular confusion, we differentiated between vowel-consonant, within-category and between-categories confusions. Most costly were confusions between a vowel and a consonent (and vice versa). Next costly were *between-categories* confusions (except of course *vowel-consonant* confusions). Within-category confusions were least costly. Confusion of a phoneme with itself did of course not result in any costs. A Viterbi dynamic programming algorithm was used to find the path of lowest cost through the matrix. The path was constrained to start at the lower left corner of the matrix and to end in the upper right corner. When the realized string equals the ideal string the optimal path follows the diagonal of this matrix. Both for square and rectangular matrices, the path did not deviate much from the 'diagonal'. We used this optimal path to map the accent from the phonemes of the ideal string on the vowels from the realized string. In case of insertions, i.e., an ideal phoneme matches two or more realized phonemes, the stress is placed on the last vowel.

For the number of syllables in a word we used a simple heuristic: it is equal to the number of vowel phonemes in the realized word.

Having described how the derived database with information about each phoneme in TIMIT speech corpus was constructed, we next decribe how it was used to select particular vowel phonemes and how the analyses were performed on these selections.

3. The experimental procedure

The experimental procedure to recognize selected vowels from different speakers can be divided into the following steps.

1. selection of the vowel material
2. performing bandfilter analysis
3. equalizing frequencies of occurrence
4. determine principal components
5. discriminant procedures
6. testing speaker normalization with neural nets

These items will be discussed in the following sections.

3.1. Selection of the vowel material

In order to asses the validity of the adaptive neural net approach on the type of speech material in the TIMIT database, we decided to start with a subset of this corpus. First of all, we limited ourselves to those vowel phonemes that are properly pronounced. Our criterium for a properly pronounced vowel is: a vowel that receives word stress. Stressed vowels in any case approximate best the vowels in the clearly uttered short sentences that were used to test the adaptive model for speaker normalization in Weenink (1993). The identity of stressed vowels as compared to unstressed ones is less subject to uncertainty and thus algorithmic identifiability has a better chance to succeed. Next, for the time being, we made some selection on the amount of data by only using the training material in the New England dialect group (dr1). This dialect

group contains enough material, 38 speakers of which 14 are female and 24 are male, to get a good impression about the merits of the procedure. We soon realized that it didn't make sense to maintain all the 20 different vowel phonemes that are listed in table 2. We selected 13 monophthongs, similar to the ones used in a comparable study of Meng & Zue (1991). These 13 selected vowels are indicated with a "+"-sign in the table. A complication, mainly of statistical nature, is the fact that the frequencies of occurrence of the vowels vary greatly which causes some problem for training a recognizer. Each speaker only realizes a subset of the complete vowel set, either because of the limited number of different vowels in the 10 sentences that the speaker spoke, or the unequal distribution of the vowels in this material, or the particular vowel inventory of the speaker's dialect. This is clearly illustrated by figure 1, where the frequency distribution of the 13 vowels in the training part of the Northern England dialect group is shown. We see large differences in the number of occurrences, the extremes /æ/ and /ʊ/ differ by an order of magnitude. We decided, somewhat arbitrary, to select maximally 4 replications of a vowel from each speaker, which amounts to maximally 152 items (38 speakers x 4 replications/speaker) of the same vowel. We checked that, despite this limitation, at least the less frequent vowels were all selected (because none of these happen to occurs more than 4 times within the vowel set of a single speaker), and, that we have sufficient instances of each vowel for data analysis. However, still the number of occurrences of the vowels were not equal after this limitation. We will cope with this later in section 3.3. This selection was further split into two parts based on speaker sex. These selections amount to a total number of 1487 stressed vowels in the training part of the Northern England dialect, 946 for the males and 541 for the females.

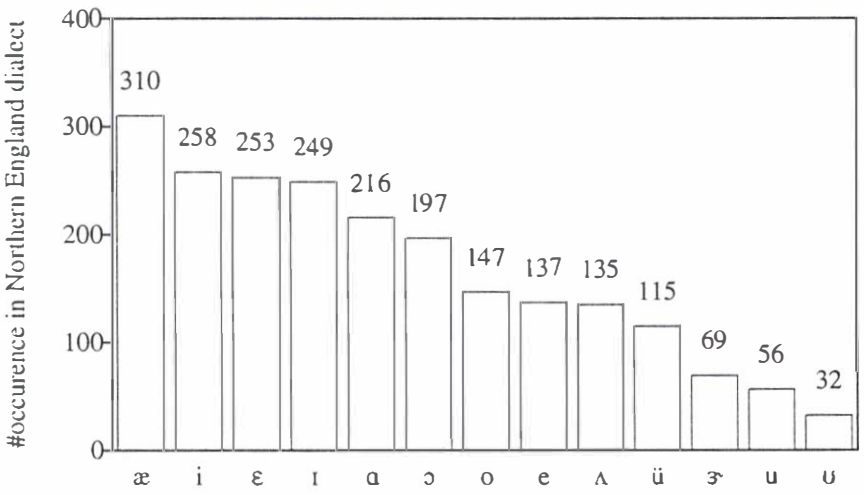

Fig. 1. Numbers of occurrence of 13 vowels in the training part of the Northern England dialect group of the TIMIT database (38 speakers). All selected vowels have word stress and are ranked from largest to smallest numbers of occurrence (exact numbers are listed above each bar).

3.2. Bandfilter analysis

We wrote a *perl* program that used our derived database and that enabled us to make flexible selections of phoneme material from the TIMIT database. The program also can make analysis scripts for the *praat* program. In this way we were able to analyze

all 1487 selected vowel segments in an automatic fashion. The analysis itself consisted of a filter bank analysis implemented in software. The filtering procedure went as follows:

```
for all selected vowels
    select a 0.050 s duration interval in the centre of the vowel
    apply a gaussian window to the interval
    calculate the Fourier spectrum of the windowed interval
    calculate the filter bank spectrum from the Fourier spectrum
    normalize the filter bank spectrum w.r.t. total power
endfor
```

The effective width of the gaussian window is approximately half its duration. This means that the effective duration of the analysis segment is approximately 0.025 s. As is implied in the above script, the analysis procedure is a static one, one frame in the centre of the vowel was analyzed. The filter bank we used consists of 16 filters at one Bark distance, each filter having approximately a 1 Bark bandwith, the first filter's centre frequency being located at 1 Bark (93 Hz), the last filter at 16 Bark (3163 Hz). The form of the filter function and other details are extensively described in Sekey & Hanson (1984). Each filter bank value was calculated from the Fourier amplitude spectrum by multiplying this spectrum with the response of the filter function centered at a particular Bark frequency and subsequently scaling the result to dB. The result of this analysis is that for each vowel we obtain an array of 16 numbers, representing the power in each filter (in dB). To remove overall power variations, these 16 values were subsequently scaled to the (arbitrary) total power level of 70 dB.

3.3. Equalizing vowel frequencies of occurrence

We used a simple procedure to compensate for the unequal number of vowel occurrences (n_i, $i = 1..13$) that still remained after limiting the maximum number of vowel occurrences per speaker to four. We first determined the maximum of the n_i (summed over all speakers). Next for all vowels with a smaller n_i the analysis data were extended with replicas: we added copies of the smaller vowel set to itself just as often as possible without exceeding the maximum n_i . This can always be done zero or more times. Next we added random draws until the new number of occurrences was equal to the maximum of the n_i. After this procedure all the new n_i were equal. In this way the male data set was extended from 941 to 1248 items and the female data set from 541 to 728 items.

3.4. Principal components

Principal components were determined for the male and female dataset separately by diagonalizing the covariance matrices. In figure 2.A and 2.B the first two principal components of the 13 average vowel spectra of the "train" part of the Northern England dialect were plotted with respect to the overall average vowel spectrum (centered). Both males and females show the same regular pattern: a clear separation between high, low, front, back and central vowels. This structure is more pronounced in the female data. The first two principal components together explain 72% and 74% of the total variance for the male and the female data respectively. The average spectra and the variances are very similar for both groups.

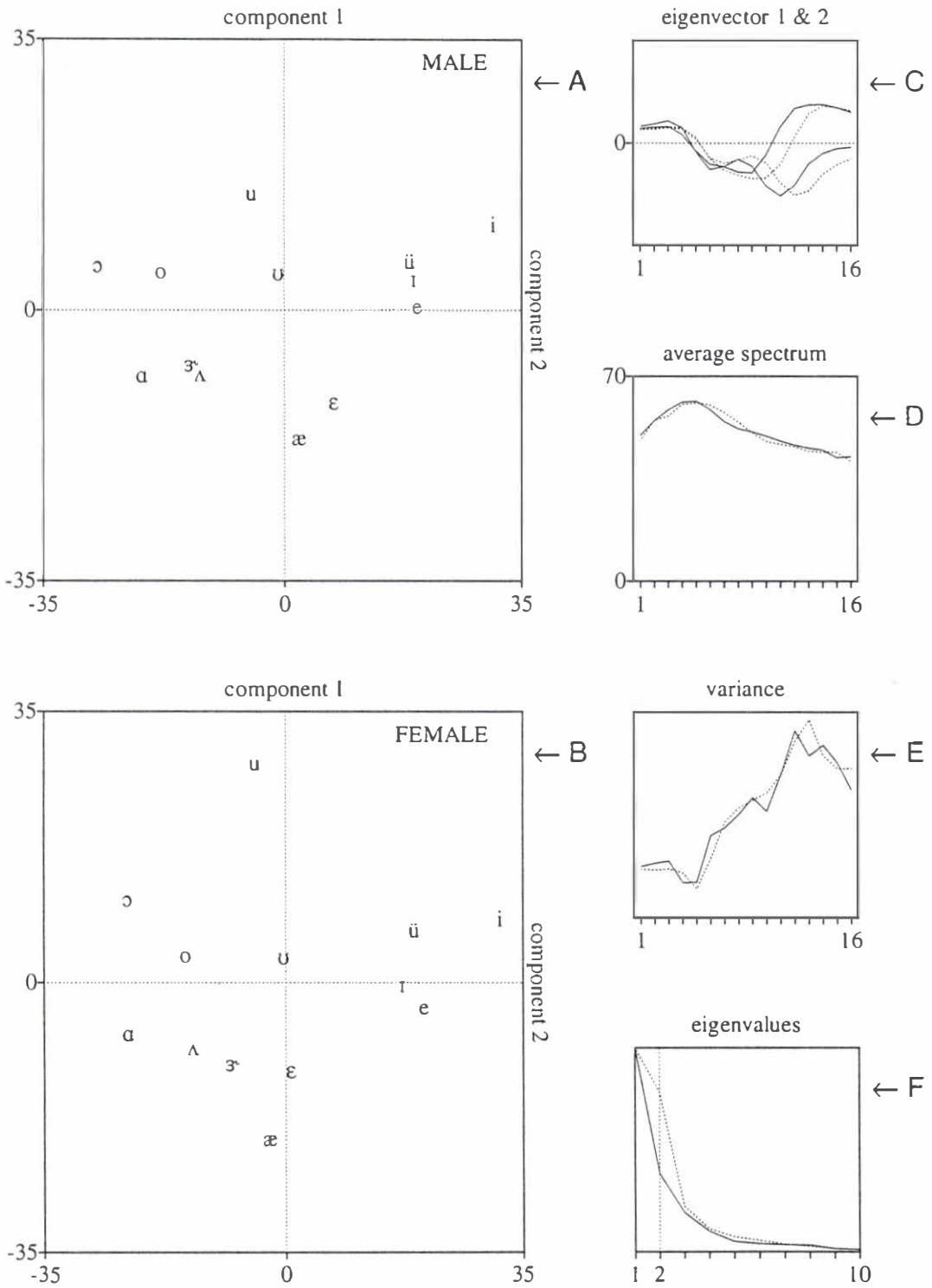

Fig. 2. Principal components and derived data for 13 vowels averaged over 24 males and 14 females in the Northern England dialect group. All vowels are from the training part of the TIMIT database and have word stress. A. The first two centered principal components for the males. B. Same as A for females. C. Eigenvector 1 and 2 both for males and for females. D. The average spectrum. E. The variance for each of the filters. F. The first 10 eigenvalues (relative scale). In C-F the data are drawn with continuous lines for males and with dashed lines for females.

3.5. Discriminant analysis

We first applied a discriminant analysis and considered this a fast means to determine a lower limit for subsequent classifications with neural nets. The discriminant analysis that we used determines only the global covariance matrix for classification. In table 3 the classification results of a discriminant analysis on the male and the female dataset are given. Considering the great contextual and idiosyncratic variability in the vowel data, 59% correct for the male data and 57% for the female data is a reasonable outcome. Trying to classifying female data with a discriminant trained on male data returns poor results, only 27% correct.

Table 3. Percentages correct classification with discriminant analysis on male and female dataset of Northern England dialect ("train"). The first column indicates what dataset has been used to train the classifier, the first row indicates what dataset has been used to test the classifier.

train \ test	male	female
male	59	27
female	22	57

3.6. Testing normalization with neural nets

We will use neural nets of the feedforward type to test the adaptation possibilities of principal components extracted from bandfilter data. This type of neural net and its classification possibilities are extensively described in Weenink (1991, 1993). When we train a feedforward neural net for subsequent classification, we hope to improve somewhat the results obtained by a discriminant analysis. Especially when a sufficient number of hidden units are involved a neural net can approximate any input-output relation. With the feedforward neural net classifier we therefore expect to increase the classification results of table 3.
Testing the neural net adaptation model consists of a number of steps.

1. Linear scaling of the first three principal components to the range (0, 1). This is a rather optimal input range for the neural net because when the initial random weights are also chosen in this range, the nodes in the hidden layer are very sensitive to change and can therefore optimally learn. We used the same linear scaling for male and for female principal components. The minimum and maximum values for the centered components always were in the range (-50, 50) so we simply added 50 and divided the result by 100 (because a principal component is a linear combination of individual filter outputs which are all smaller than 70 dB its value may exceed 70 dB).

2. Training the neural net. The total number of weights and biases, i.e., the number of parameters, to be trained for a neural net with N inputs, H hidden units and M outputs is $H(N+M+1)+M$. In Weenink & Pols (1993) best results were obtained with three (linearly scaled) formant frequency inputs and 3 hidden units. We will now use 3 scaled principal components as inputs and also three hidden units. The only difference is that we now have 13 outputs rather than 9. A minimum squared error cost function is used during the training (Weenink, 1993).

3. Adaptation to the test set. Here adaptation is modelled as a variation of bias in a trained neural net. As an example consider a net with topology (3, 3, 13) that has learned the associations between series of 3 principal components and corresponding vowel categories from the *male* data set. During this learning process 64 parameters

had to be optimized. Adaptation to the *female* data set is then a learning process in which only three biases, the ones from the hidden nodes, are permitted to change. Despite the fact that instead of 64 there are only 3 parameters to optimize, this task is not a trivial one because the minimization process often halts in a local minimum and therefore does not reach the optimum. We therefore had to restart the minimization procedure several times with different random values for the weights until satisfactory results were obtained. Table 4 shows compiled results of the bias adaptation from the earlier study with formant frequency values (Weenink & Pols 1993).

4. Classification of the test sets. The neural net uses a winner-takes-all classification strategy.

Table 4. Adaptation performance of a feedforward neural net with topology (3, 3, 9) trained on formant frequency data. The first column indicates what dataset has been used to train the classifier, the first row indicates what dataset has been used to test the classifier. The off-diagonal numbers in italics were obtained after bias adaptation and the numbers in parenthesis before. The table is a compilation from table 1 & 2 in Weenink & Pols (1993). The male data set is from Van Nierop et al. (1973), the male data set is from Pols et al. (1973).

train \ test	male	female
male	89	*88* (78)
female	*79* (68)	95

Table 5. Adaptation performance of a feedforward neural net with topology (3, 3, 13) trained on the first three principal components from a bandfilter analysis of stressed vowel segments from male and female speakers in the Northern England dialect ("train" part only). The first column indicates what dataset has been used to train the classifier, the first row indicates what dataset has been used to test the classifier. The off-diagonal numbers in italics were obtained after bias adaptation and the numbers in parenthesis before.

train \ test	male	female
male	49	*50* (44)
female	*48* (37)	49

In table 5 we have collected the results of the test of the adaptation model with a neural net trained on the male and female data. On the diagonal of this classification table the results are given for the condition that the data set used for the training was also used as the data set for the testing. Compared with table 4, we note the relatively low score obtained when the test set is equal to the train set. This low score with three principal components is of course no surprise considering the scores in table 3, based on the discriminant analysis where the complete 16 dimensional input was used. The percentage of the variance explained by the first three components is approximately 83% for both males and females. Probably the percentage correct classification with p components scales with the percentage explained variance by these p components. Although the adaptation procedure as such seemed to work reasonably well still the classification scores are rather low.

A further test with the complete bandfilter spectrum instead of 3 principal components showed improvement on the classifcation results when the train set was used as test set, however, adapation performance was worse. The results of this

classification can be found in table 6. Two different neural net topologies were used, (16, 0, 13), no hidden units at all which amounts to a non-linear discriminant analysis, and, (16, 16, 13). The inputs for the neural net are the 16 filter bank values scaled to the range (0, 1) by dividing each value by 70. When we compare the numbers in this table with the numbers in table 3 we see a significant improvement in classification performance by using a neural net rather than discriminant analysis.

It seems that a classifier that only uses static data, i.e., one central measurement frame per vowel as we have done above, has its limit at approximately 70-80% correct classification. Maybe some percents better classification can still be obtained with another even more optimal parametrization of the signal. Using dynamics might further improve the score (Huang, 1991). In order to asses how much it would help to include time variable information, we performed another filter bank analysis on the material. This time we analyzed for each vowel three consecutive segments at 0.02 s intervals, the second one being at the middle of the vowel. In this way the second segment is positioned at the same place as in the static analysis. All filter bank values were scaled in the same way as before. This resulted in 48 (=3 x 16) values for each vowel. These 48 values were used as input to neural nets with zero and 16 hidden units. The two last rows in table 6 give results for the classification of these dynamic data, please note the significant improvements in vowel identification scores for these dynamic data. This indicates that enough spectro-temporal information is present in these 60 ms vowel segments in various contexts to classify them well. However, we also have to conclude that although the bias adaptation model showed some effect, even for a limited number of principal components, its potential extendibility is limited.

Table 6. Percentages correct vowel classification with a feedforward neural net of different topologies on the filter bank values from stressed vowel segments from male and female speakers in the Northern England dialect (only "train" part). The first column indicates the topology (number of inputs, number of hidden units, number of output units). Test and train set were the same. The last two rows are results for dynamic data (three consecutive frames).

topology	male	female
(16, 0, 13)	65	68
(16, 16, 13)	77	79
(48, 0, 13)	83	85
(48, 16, 13)	88	90

4. Discussion

The results of the normalization procedure as described in Weenink & Pols (1993) for fixed context vowel data could not completely be reproduced with bandfilter representations from much more variable speech data. This denotes that the bias normalization model was too optimistic. With formant frequency values the effect of adapting the bias in the neural net model results in an almost linear scaling of these formant frequency values. In a way this amounts to vocal tract length scaling. Because formant frequency values and principal components have a high correlation, this linear scaling did also work with principal components. In fig 2C we see that the first

principal component for the male and the female data are, more or less, related by a translation. This also happens to be the case for the corresponding second principal components. This can also be interpreted as scaling. Although classification results improve as the number of inputs increase, the neural net bias adaptation becomes worse. However, even if the model should have worked perfectly for all input dimensions, i.e., always resulted in significant better recognition scores after adaptation, than, properly speaking, we still cannot speak of a real adaptation model. This type of adaptation or normalization is, after all, supervised and a posteriori. In this respect the adaptation model behaves as all the other a posteriori formant frequency normalization methods that can be found in the literature (e.g., Gerstman, 1968; Lobanov, 1971; Miller, 1989; Pols et al., 1973; Van Dijk, 1981). The real challenge is in finding a method that works in real time. It may be clear that the supervised networks of the feedforward type that we used can never accomplish this formidable task. Markov models, although very powerful, cannot accomplish this task either. What we really need is adaptation in real time, i.e., as soon as the input data arrives. A necessary precondition for this is, among other things, self-organization. In Grossberg (1988, section 17) a long list of necessary computational qualities are discussed that such systems should have. ART (Adaptive Resonance Theory) neural network models are potentially well suited for the job (Grossberg, 1988; Nigrin, 1993). These models and their implementation in the speech domain will be the topic of future investigation.

Acknowledgement

The author wants to thank Louis Pols for his critical and constructive comments during this study.

References

Bloothooft, G. (1985), *Spectrum and timbre of sung vowels*, Thesis, Vrije Universiteit, Amsterdam.

Boersma, P. & Weenink, D. (1996), *Praat, a system for doing phonetics by computer*, version 3.4, Institute of Phonetic Sciences University of Amsterdam, report 132.

Chan D. et al., "Eurom - a Spoken Language Resource of the EU", *Proc. Eurospeech '95*, Madrid, Vol 1, 867-870.

Den Os, E.A., Boogaart, T.I., Boves, L. & Klabbers, F. (1995), "The Dutch Polyphone Corpus", *Proc. Eurospeech '95*, Madrid, Vol 1, 825-828.

Gerstman, L.H. (1968), "Classification of self-normalized vowels", *IEEE Trans. Audio Electroacoust.*, **AU-16**, 78-80.

Grossberg, S. (1988), Nonlinear neural networks: principles, mechanisms, and architectures, *Neural Networks* **1**, 17-61, reprint in: Carpenter, G.A. & Grossberg, S. (eds.), *Pattern recognition by self-organizing neural networks*, A Bradford Book, The MIT Press, 1991.

Huang, C.B. (1991), "An Acoustic and Perceptual Study of Vowel Formant Trajectories in American English", *Research Laboratories Electronics Massachusets Institute of Technology*, Technical Report No. 563.

Lamel et al. (1994), "The Translanguage English Database (TED)", *Proc. ICSLP '94*, 1795-1798.

Lamel, L.F., Kassel, R.H. & Seneff, S. (1986), Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus", *Proc. DARPA Speech Recognition Workshop*, Report No. SAIC-86/1546, 100-109.

Lobanov, B.M. (1971), "Classification of Russian vowels spoken by different speakers", *J. Acoust. Soc. Am.* **49**, 606-608.

Meng, H.M. & Zue, V.W. (1991), "Signal representation comparison for phonetic classification", *IEEE Proc. ICASSP '91*, Toronto, 285-288.

Miller, J.D. (1989), "Auditory-perceptual interpretation of the vowel", *J. Acoust. Soc. Am.* **85**, 2114-2134.

Nigrin, A. (1993), *Neural Networks for Pattern Recognition,* A Bradford Book, The MIT Press.
Pols, L.C.W., Tromp, H.R.C. & Plomp, R. (1973), "Frequency analysis of Dutch vowels from 50 male speakers", *J. Acoust. Soc. Am.***53**, 1093-1101.
Pols, L.C.W., Van der Kamp, L.J.Th. & Plomp, R. (1969), "Perceptual and physical space of vowel sounds", *J. Acoust. Soc. Am.* **46**, 458-467.
Sekey, A & Hanson, B.A. (1984), "Improved 1-Bark bandwidth auditory filter", *J. Acoust. Soc. Am.* **75**, 1902-1904.
Sulter, A.M. & Schutte, H.K., Voice Research Lab, Department of Oto-Rhino-Laryngology, Oostersingel 59, P.O. Box 30.001, 9700 RB Groningen.
Van Dijk, J.S.C. (1981), "How to normalize your own vowel system", *Proceedings of the Institute of Phonetic Sciences Amsterdam* **6**, 67-72.
Van Nierop, D.J.P.J, Pols, L.C.W. & Plomp, R. (1973), "Frequency analysis of Dutch vowels from 25 female speakers", *Acustica* **29**, 110-118.
Weenink, D.J.M. & Pols, L.C.W. (1993), "Modelling speaker normalization by adapting the bias in a neural net", *Proceedings Eurospeech '93*, 2259-2262.
Weenink, D.J.M. (1993), "Vowel classification with neural nets: a comparison of cost functions", *Proceedings of the Institute of Phonetic Sciences Amsterdam* **17**, 1-11
Weenink, D.J.M. (1991), "Aspects of neural nets", *Proceedings of the Institute of Phonetic Sciences Amsterdam* **15**, 1-25.