formance difference. The major difference between these cases was that the current networks, due to their number and the time required for their training, had been constructed with fewer hidden units -- and consequently fewer tunable parameters. For this reason, the experiment was repeated with a network with the same number of hidden units as had been used in the initial experiment. The superfluous I/O units for absent speakers and phones were, however, omitted. There were *n* input units, eight units in the first hidden layer and fourteen in the second, and nine output units. The biased network had twenty speaker ID inputs available via two hidden units in a separate layer.

These networks had the training[19] set classification performance given in Table 30. The unbiased raw input (48 input) performance was almost the same as in the preliminary experiment, although in this case the biased performance was slightly better, not worse, than the unbiased performance[20]. As before, the improvement in recognition performance was more

	Input Dimension													Raw
Case[a]	1	2	3	4	5	6	7	8	9	10	11	12	13	48
Bias	66.9	74.0	77.5	78.4	79.1	80.4	81.3	80.7	81.7	81.1	82.1	82.0	82.4	85.7
Simple	59.8	66.5	71.3	72.7	74.9	75.8	76.3	76.4	77.4	77.7	79.9	79.1	79.3	84.0
% Redn	17.7	22.4	21.6	20.8	17.0	19.2	21.2	18.4	19.2	15.4	11.1	13.6	14.9	10.9

Table 32: Improvement in Recognition performance from Speaker Bias for RMSpell Vowel data projected to various dimensions.

a. This table gives the result for randomised pattern presentation only.

substantial when there was less speech data available for the classifier to work with. This effect is demonstrated in the following graph which displays the percentage error reduction available from speaker ID, plotted against input dimension, for both the networks with wide (8,14) hidden layers and the network with narrow (4,7) hidden layers:

5.8.3. Conclusions from Peterson Barney/RMSpell Comparison

These experiments were prompted by the observation that while the SVC-generating speaker models had been of unspectacular utility in the French Digits recogniser (§), and while speaker ID inputs had produced no improvement at all in the pilot RMSpell Experiment (§5.4), experiments in [watrous93] that ought to have served as a model for this kind of speaker "adaptation", using information about speaker identity, had shown impressive gains.

---

19. it is important to note that these results, with diminishing returns for speaker ID with increasing input data, are training results, since, while this is the practice of the experiments [watrous93] on which the current work was modelled, it is not, as we shall see later a wise one, if we wish to estimate the usefulness of speaker ID on real tasks. Even if the practice is justified when the data-sets available are very small, it would have been wise to abandon it immediately during the move to a larger data-set.

20. Why the effect if speaker ID bias on training set performance was not visible in the pilot run remains somewhat of a mystery. Since the biased performance of the presence network was better than the unbiased performance in the pilot, it is possible that the presence of spurious output units, trained to produce zero outputs, caused the network to find an initial solution to the problem that ignored the speaker IDs, and that could not be escaped during further training.

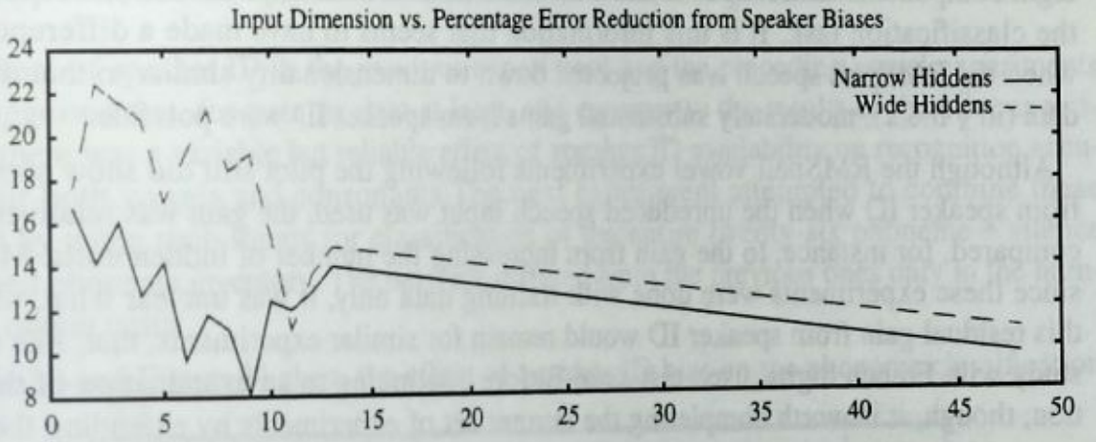

Figure 33: Error reduction vs. dimension onto which the three frames of input data were projected. The "Narrow Hiddens" case was for a network with four and seven hidden units in the first and second hidden layers, respectively. The "Wide Hiddens" network had eight and fourteen units in the corresponding layers. Especially in the case of the "Wide Hiddens" network, speaker biases produce a more substantial error reduction when less information is available from the speech directly.

The experiments in §5.6.2 showed that when speaker voice codes were derived from the Peterson and Barney vowel data (PB), they were effective at improving recognition performance on the task, although not as effective as perfect speaker information; if Peterson and Barney data could be used as a model for adaptation at all, it could be used as such with the sort of speaker "ID" provided by the SVCs investigated in this thesis. The almost complete failure of the pilot experiments with the more realistic Resource Management Spell-Mode (RMSpell) data-set suggested, therefore, that either the Peterson Barney data was not a good data set to work with as a model of real speech, or that there was something about the training done on the two data sets that made speaker adaptation work well with one, but not the other.

The experiments in §5.6.1 compared second order nets of the sort Watrous had used in [watrous93] with the more conventional first order nets used in the bulk of the experiments in this thesis. In no case did the second order nets provide any advantage over conventional ones. In fact, a conventional backpropagation network with speaker ID as additional input, with a 98.4% classification accuracy, outscored the best performance (97.5%) reported in Watrous's paper for a second-order net. Network architecture was clearly not the cause of the performance differences.

Moreover, the difference in input representation between the formant values used in the PB experiments and the spectra used for RMSpell did not appear to be an impediment to adaptation, at least for conventional nets, since converting the PB formant data to a spectral representation, while holding the amount of information constant (in §5.8.1) only slightly decreased recognition accuracy.

What did appear to make a difference was the amount of information available in the raw speech data. The PB data, containing only two formant values for each vowel, furnished only a very small, measured, amount of information that could be used for vowel identification. The speech used in the RMSpell experiments, by comparison, contained, along with information about formant position, a good deal of other information in the sixteen to forty-

eight components of the input vector. Some of this information, presumably, was relevant to the classification task. It is this information that seems to have made a difference, since when the RMSpell speech was projected down to dimensionality similar to that of the PB data (in § 5.8.2), moderately substantial gains from speaker ID were possible.

Although the RMSpell vowel experiments following the pilot still did show some effect from speaker ID when the unreduced speech input was used, the gain was relatively minor compared, for instance, to the gain from increasing the number of hidden units. Moreover, since these experiments were done with training data only, it was unclear what, of any, of this residual gain from speaker ID would remain for similar experiments, that, like the pilot study with French digits, used test data. Before continuing to an examination of that question, though, it is worth completing the current set of experiments by extending the task to recognition of constants, and to speech with a full phoneme inventory.

5.8.4. Speaker ID used with Consonants

Since it had been possible to demonstrate a small amount of utility in using speaker IDs to inform the recognition of vowels from the RMSpell database, but none at all for the all-phone recogniser in the pilot RMSpell experiment in §5.4 it seemed possible that speaker ID was unhelpful for consonant recognition, and that the rather small effect that had been shown for vowels had been diluted by the presence of consonants in the all-phone case. This hypothesis was also consistent with the widely held notion that voice quality or voice personalty is expressed more strongly in vowels.

	Input Dimension												Raw
Case	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**48**
Bias	41.9	68.0	68.5	73.1	73.0	75.6	76.3	76.0	76.6	77.5	77.1[a]	78.3	79.8
Simple	31.2	58.8	63.2	65.4	67.3	68.7	70.4	70.2	71.8	70.6	71.8	72.1[b]	75.1
% Redn	15.5	22.4	14.3	22.2	17.4	22.0	20.0	19.5	17.0	23.4	18.8	22.1	19.2

Table 33: Improvement from speaker bias in recognition performance for RMSpell Consonant data projected to various dimensions.

a. Training stopped after 5 600 epochs for this case. Others trained for 5 800 epochs.

b. Training stopped after 5 400 epochs for this case

A network like the one used in the vowel case, but with output units corresponding to the seventeen consonants, was trained to classify consonants extracted from the same sixty sentences used in the vowel experiments. Surprisingly, as shown in Table 30, the effect of speaker ID on recognition accuracy was actually stronger than it had been for vowels, and, in contrast to the vowel case, fairly stable across input dimensionalities, remaining present even when all the information in the three frames of speech was available.

5.8.5. Speaker ID used with Vowels, Consonants and Silence.

The effect of speaker ID in the previous experiment and the preceding vowel experiments had suggested that, for training data at least, and contrary to the results of the pilot experiment, there was a variable but reliable effect of speaker ID availability on recognition accuracy for both vowels and consonants The next experiment attempted to combine these results by using such inputs for classification of the entire twenty-six phoneme + silence RMSpell phoneme inventory. The network differed from the previous ones only in the number of output units.

Table 30 and Figure 34 show the effect of speaker ID bias on the phoneme classification

Case	Input Dimension						Raw
	1	2	4	6	9	13	48
Bias	48.7	65.7	71.9	74.8	75.2	77.0	78.1
Simple	44.6	59.9	68.2	71.9	73.6	74.9	77.0
% Error Reduction	7.5	14.5	11.9	10.4	6.0	8.3	4.7

Table 34: Improvement in Recognition performance from speaker ID bias for RMSpell data. This experiment used the full phoneme set. As usual, three frames of input were projected to various dimensions. The networks had 8 and 14 hidden units respectively in two hidden layers.

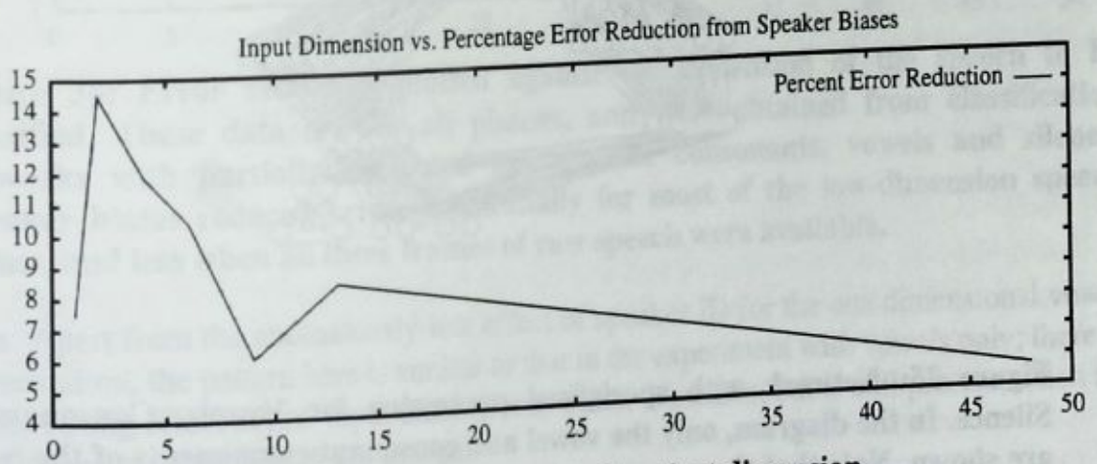

Figure 34: Error reductions of Table 30 plotted against dimension.

performance of this recogniser. Speaker biases helped considerably less in this case than in the recognisers specialised for vowels or consonants. Analysis of confusion matrices for the forty-eight input case suggested that the speaker ID biases were allowing the network to improve vowel recognition, but that this was achieved at the cost of reduced accuracy on consonants. While the number of vowels correctly classified increased with the bias, the number of correctly classified consonants decreased. These changes are enumerated in Table 35. Since, in the previous experiment, biases were more effective at improving the recogni-

tion accuracy for consonants than for vowels, it appeared that the effects of bias on vowels and consonants might be incompatible.

Vowel	AA	AH	AX	AY	EH	EY	IY	OW	UW								
	30	-	-	15	44	36	-	-3	20								
Consonant	B	CH	D	F	JH	K	L	M	N	P	R	S	T	V	W	Y	Z
	-13	-	5	-1	-7	-	41	-20	-24	-	-12	13	-22	-	1	18	-

Table 35: Speaker biases appear to affect vowels and consonants differently. This table summarises the change in the number of correctly classified frames for each phone when speaker information is available. In a unified network, improvements in the number of correctly classified inputs for vowels were largely offset by decreases for consonants. Cells containing "-" exhibited no change in recognition accuracy when speaker biases were used.

5.8.6. All Phones, split nets - Separate Vowel, Consonant and Silence Nets

In an effort to prevent any such interference, it was worthwhile to experiment with a system in which the network was divided up, below the level of the output units, into separate recognisers for vowels, consonants and silences, all of which received the same input. The architecture is illustrated in Figure 35. This splitting of the network was intended to force

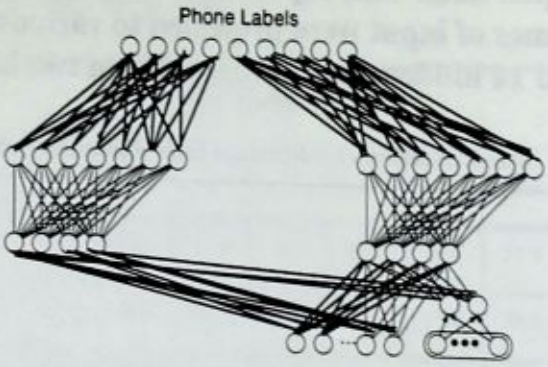

Figure 35: Network with specialised processing for Vowels, Consonants and Silence. In the diagram, only the vowel and consonant components of the network are shown. Note that the networks share a common input, and a common output layer, but that the hidden units are specialised for vowels and consonants (shown) and silence (omitted from the diagramme).

the lower layers of the network to adapt, if at all, in ways relevant to the speech sound being specialised for. It is important to note that in this architecture, since the three networks were still competing with each other to produce the highest output, the training was fully discriminant across the entire phone set.

Table 30 and Figure 36 give the effect of biasing the network as in the previous experi-

Case	Input Dimension						Raw
	1	2	4	6	9	13	48
Bias	48.7	65.9	73.3	76.3	77.7	80.1	81.1
Simple	44.6	59.9	68.6	72.6	73.7	76.6	79.2
% Error Reduction	7.4	15.1	14.9	13.3	15.1	15.0	9.1

Table 36: Improvement in Recognition performance from Speaker Bias for RMSpell data for all phones projected to various dimensions, using split nets (8,14 hiddens)

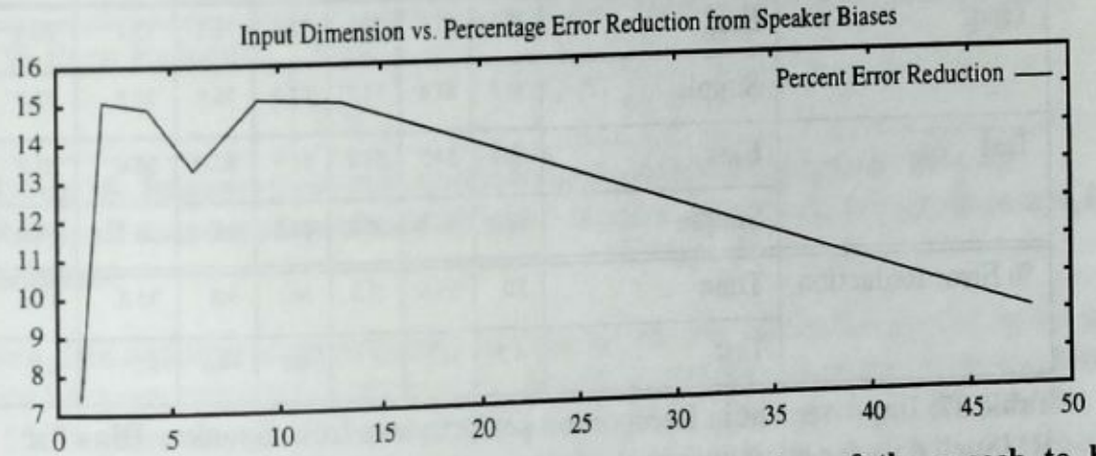

Figure 36: Error reduction plotted against the dimension of the speech to be classified. These data are for all phones, and were obtained from classification networks with partially separated modules for consonants, vowels and silence. Speaker biases reduced error substantially for most of the low-dimension speech inputs, and less when all three frames of raw speech were available.

ments. Apart from the anomalously low effect of speaker ID for the one dimensional vowel representation, the pattern here is similar to that in the experiment with vowels only; there is a rather strong effect at low input dimensions, which decreases, but is still present, when all three frames of input speech are present.

Discussion

Although a true speaker model was not being used, and although the experiments with speaker models for the Peterson and Barney data-set (§5.6.2) had suggested that the effect of such models would be smaller than that of the idealised speaker-ID "model" used here, and although the effect of bias was less than for the Peterson Barney data, it had, at least, by this stage, been possible to obtain an effect of speaker bias on phoneme recognition for all phones, on *training data*. This matched, more or less, the status of Watrous' work with Peterson & Barney data. Of course, if adaptation by biasing, whether by ideal inputs or by speaker models, was to be of any practical use, it would have to improve performance on

untrained speech as well as on the training set, and, in the case of speaker models, on untrained speech from novel speakers. The next experiment was designed to test for such an effect.

5.8.7. Testing the Speaker ID effect for Generalization

In order to see how well the effect of bias would generalise to other speech from the same speakers, the experiment was repeated. This time three utterances of training speech and one of testing speech were used for each speaker. As before, the networks were split into vowel, consonant and silence subnetworks internally. Table 30 gives the recognition accuracies for

Case		Input Dimension						Raw
		1	2	4	6	9	13	48
Train	Bias	47.1	67.4	76.9	72.8	79.1	75.1	76.1
	Simple	44.3	61.4	73.8	67.5	76.9	70.5	72.9
Test	Bias	36.9	54.2	58.0	61.9	62.0	66.0	65.8
	Simple	38.0	56.0	60.0	62.2	63.6	64.1	64.9
% Error Reduction	Train	5.0	15.6	11.9	16.3	9.6	15.8	11.6
	Test	-1.7	-4.2	-5.1	-0.8	-4.6	5.2	2.7

Table 37: Improvement in Recognition performance from Speaker Bias for RMSpell data for all phones projected to various dimensions, using split nets. In this case, three utterances of training and one of testing data were used for each speaker.

a number of input dimensions.

The difference between training and test set performance was wider than one might have expected. In the unbiased condition, the misclassification rate increased almost 30% for the test set, suggesting that the network might have been overtrained. More interestingly, there was an odd effect of the speaker ID information on the lower dimensionality versions of the test set (projected along the same axes as the training set). It would seem, at these dimensionalities, there was no generalisation at all of the large effect that speaker ID had on the training set. Even in the higher dimensional cases, the effect of speaker ID on the test set was rather small.

Given that these networks seemed to be failing to generalise well both with respect to the simple classification task itself and with respect to the effect of speaker ID biases on performance, it appeared that it would be worth while to construct a training set with more data and to run this experiment again.

More Training data

This experiment used six training utterances for each speaker, but was otherwise identical to the previous experiment. Table 30 displays the performance of this network in the usual

Case		Input Dimension						Raw
		1	2	4	6	9	13	48
Train	Bias	43.9	62.5	70.0	73.0	74.4	75.7	78.3
	Simple	41.7	58.9	66.5	69.7	71.8	73.8	76.2
Test	Bias	49.1	64.9	69.4	72.7	74.1	74.7	76.5
	Simple	49.3	62.5	68.8	70.9	73.2	74.2	74.0
% Error Reduction	Train	3.7	8.8	10.6	11.1	9.3	7.5	8.7
	Test	-0.2	6.6	2.2	6.0	3.2	2.1	9.6

Table 38: Improvement in Recognition performance from Speaker Bias for RMSpell data for all phones projected to various dimensions, using split nets (8,14 hiddens)

manner. The addition of more training data per speaker was successful at reducing the difference between training and testing set on the raw recognition task. In fact, there was better than 100% generalisation for some of the lower dimensionality results, probably just due to chance differences between the composition of the training and testing set. Figure 37. shows

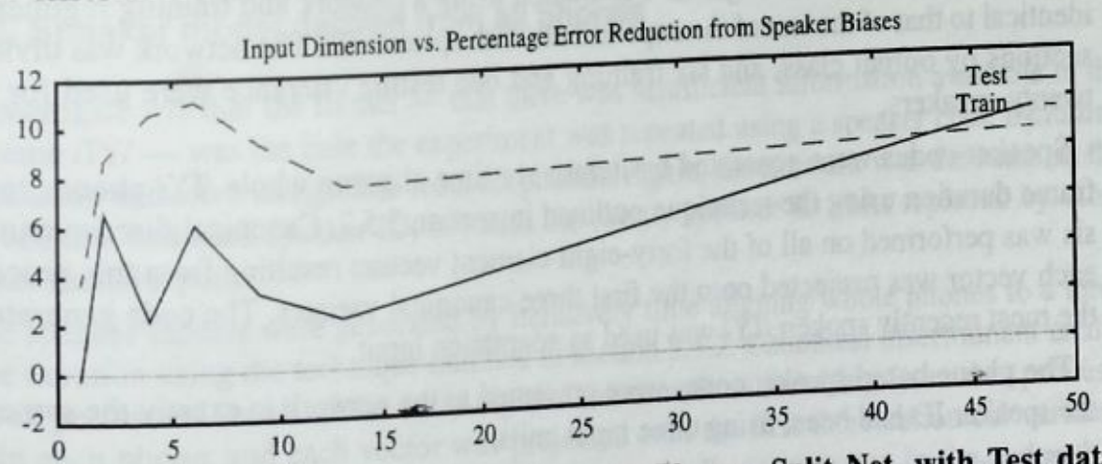

Figure 37: Error reduction vs. dimension - All Phones, Split Net, with Test data, trained on twice as much data as the previous experiment.

the percentage error reduction from Speaker ID biases, for this data. Although there was little sign of overtraining in the recognition performance, the effect of speaker ID on performance for the test set was still generally lower than for the training set. The speaker dependent characteristics of the data-set that the network is able to learn seem to be less robust than the speaker independent ones.

Although, again, in this experiment, as in the experiments where only training set classification performance was measured, there was some evidence that having access to information about speaker identity had a useful effect on recognition performance, even in these cases of perfect speaker information, the effect was slight. It was certainly too slight to justify any particular confidence that these sorts of connectionist classifiers, or full recognisers built on top of them, would furnish a practical tool for measuring the performance of competing imperfect speaker models, where the effects on classification performance, if present at all, would be more or less guaranteed to be smaller.

5.9. Using a Speaker Model in Recognition

Despite lack of a particularly promising outlook for the activity, an attempt was made to apply speaker models derived from speech, rather than speaker IDs, to the task of improving speech recognition. The speaker models used here differed somewhat from most of those described in the previous chapter, since not all of the work described there preceded these speech recognition experiments.

5.9.1. Information derived from only the phoneme /IY/.

Since full speaker models are somewhat complex, it is difficult, when using them, to have an accurate understanding how much information had been presented to the recogniser. Following a suggestion form Bridle [Bridle, 1993, personal communication], it seemed worthwhile to see whether any effect on recognition performance could be obtained from a single phoneme. The phoneme /IY/ was used for this purpose, since it is relatively frequent, and because, since it is a vowel, it should contain information about voice personality.

In this experiment, a recogniser was trained using a network and training regimen nearly identical to that of the last of the experiments with speaker ID: The network was divided into sections by output class, and six training and one testing utterance were used for each of twenty speakers.

Speaker codes were generated by iteratively time aligning whole /IY/ phones to a three frame duration using the technique outlined in section 3.5.2. Canonical discriminant analysis was performed on all of the forty-eight element vectors resulting from this process, and each vector was projected onto the first three canonical variates. The code generated from the most recently spoken /IY/ was used as adaptation input.

The phone based speaker codes were presented to the network in exactly the same way as the speaker ID had been, using three input units.

The performance of this network after training is given in Table 30.

There was no consistent improvement in classification accuracy for networks given the /IY/ bias.

Case		Input Dimension						Raw
		1	2	4	6	9	13	48
Train	Bias	42.4	58.9	66.1	68.3	70.4	71.3	73.5
	Simple	41.6	58.6	65.7	68.2	69.3	71.1	74.2
Test	Bias	49.5	61.9	68.2	70.9	71.8	71.7	72.8
	Simple	49.2	61.8	68.6	70.2	71.2	72.2	73.6
% Error Reduction	Train	1.4	0.7	1.2	0.3	3.6	0.7	-2.7
	Test	0.6	0.3	-1.3	2.3	2.1	-1.8	-3.0

Table 39: Improvement in Recognition performance from Speaker Bias for RMSpell data for all phones projected to various dimensions. In the Bias condition, a speaker code derived solely from the information in instances of the phoneme /IY/ was available to the network.

Although it is hard to doubt that there is meaningful information about speaker ID in the /IY/ phones, it was either insufficient to make a significant difference to the performance of the phone classification task, or it was information that the network could get from elsewhere in the three frames of speech it was attempting to classify.

5.9.2. Speaker model derived from all phones

To investigate whether the former — that there was insufficient information available in the phoneme /IY/ — was the case the experiment was repeated using a speaker code including information extracted using CDA from all phones. Again, the network was the same as the network that had used speaker ID, but with the twenty speaker ID units replaced by three speaker model inputs.

The speaker models were generated by iteratively time aligning whole phones to a three frame duration using the technique outlined in section 3.5.2. Canonical discriminant analysis was performed on all of the forty-eight element vectors resulting from this process, within each phone, and each vector was projected onto the first three canonical variates. These phone models were then inserted into an overall input vector in the order they appeared in the database, resulting in a sequence of eighty-one element vectors[21]. These

21. Three elements per phone, multiplied by twenty-seven phones.

vectors were, again, subject to a CDA by speaker, yielding a three element speaker code.

Case		Input Dimension						Raw
		1	2	4	6	9	13	48
Train	Bias	41.8	59.4	67.1	70.5	72.3	73.7	76.1
	Simple	41.7	58.9	66.5	69.7	71.8	73.8	76.2
Test	Bias	49.5	62.8	68.8	71.8	73.1	73.5	75.3
	Simple	49.3	62.5	68.8	70.9	73.2	74.2	74.0
% Error Reduction	Train	0.2%	1.2%	2.1%	2.6%	1.8%	-0.4%	-0.4%
	Test	0.4%	0.8%	0%	3.1%	-0.4%	-2.7%	5.0%

Table 40: Improvement in Recognition performance from Speaker Bias for RMSpell data for all phones projected to various dimensions. In the Bias condition, three units are used to present a CDA based SVC derived from previous speech.

that changed on every phone boundary. This code was fed into the three speaker model input units of the recogniser.

Recognition results for the training network are given in Table 30. There was no consistent gain in recognition performance from using the SVC, either for training or testing data. This is consistent with the prediction that, since the SVC is, almost certainly, only an imperfect approximation to speaker ID, which itself produced only a small change in recognition performance, any effect of the speaker model based biases would be difficult to detect.

5.10. Conclusion

Information about speaker identity, whether derived from a speech signal or supplied directly, can contribute to improving recognition performance, and these gains can be obtained even within relatively simple connectionist architectures. However, in recognisers using powerful classifiers and sufficiently rich input representations, the gains can be less than spectacular. When a classifier for all phonemes was applied to the RMSpell database, in fact, speaker information derived from speaker models had an effect that was too weak to be detected, if it had an effect at all.

The work described in this chapter to bridge the gap between the very strong effect of speaker information in previous work with the Peterson and Barney vowel database and the rather weak effect of similar information used in realistic tasks, suggested an important *caveat* for speech researchers: If simplified speech signals are used, care must be taken to ensure that speaker adaptation is providing information that cannot be obtained from the speech signal over short durations, rather than simply replacing information which has been lost from the speech signal by the chosen input coding.

These experiments also served to expose some weaknesses in speech recognition as a vehicle for comparing speaker models. Clearly, the major difficulty was the fact that building recognisers that can be adapted by biasing or other forms of modulation, as opposed to adaptation by retraining, is a difficult problem in itself, quite apart from the matter of whether or not that modulation is derived from previous speech, and how. Even if that problem were solved, though, and a suitable recogniser could be built, it would still be true that recognition is a rather opaque task. If the speaker models were successfully applied, the only evidence would be an increase in classification or transcription accuracy, perhaps broken down by speech unit. This would offer a poor vehicle for investigation of such questions as whether the speaker model is perceptually relevant.

To provide a more transparent application, where the influence of changes in speaker model would be directly visible (or, rather, audible), and to investigate another application area for voice models, a series of experiments in mimicry synthesis by voice transformation were carried out. These are described in the following chapter.

Chapter 6. Synthesis By Voice Transformation

Speech recognition had not proved to be as good a test bed for speaker modelling as one might have hoped, and, even if it had been, the effect of the speaker models would have been opaque. If a clear effect had been present, it would only have been visible as a change in a recognition score, making it difficult to understand how the speaker model was encoding speaker differences. For both these reasons, but chiefly the former, a new application where the effects of modelling would be more transparent was chosen. That application is the transformation of one voice into a set of other voices, with the target voice being described by the speaker model.

Although the main purpose here was to provide an environment for evaluating the speaker modelling system, and although none of them would be adequately served by the system as it stands, one can imagine practical uses for this sort of mimicry synthesis:

- In speech synthesis, it is important to provide a voice that the listener finds agreeable; It would be very convenient to be able to select the voice one wants one's computer to use simply by playing it a sample of that voice.
- In Speech-to-Speech translation systems such as Janus [waibel91], it would be desirable to produce the translated speech in the original speaker's voice, both for aesthetic reasons, and because of the obvious utility of being able to distinguish multiple speakers in conference calls and other meetings.
- Using an inverted form of voice transformation, it would be useful for automated voice-response systems (such as voice-mail prompts) to be able to utter user-recorded segments (such as the mailbox owner's name) in the same voice as the standardised prompts, rather than the voice of the original speaker.
- Works of interactive fiction would be enhanced if the characters inhabiting them could be given a variety of realistic voices.

6.1. Other voice transformation systems

Despite the existence of these potential applications, there has been relatively little work done in this area. The work that has been done has focussed entirely on the problem of transforming the speech of a single source speaker into that of a single target speaker, although voice normalisation, the attempt to transform the speech of a number of speakers into that of a single reference speaker, for use in speech recognition, could be considered closely related.

Transformation with manual intervention

One of the earliest approaches to the problem of voice transformation was directed more towards the central problem of this thesis, understanding the differences between speakers.

Childers et al. [childers85,childers89] looked at converting single sentences from a single pair of speakers, one male and one female, into the voice of the other, using an analysis-synthesis system. The system allowed them to modify the pitch, spectral expansion, and glottal pulse shape of the source speech to more closely match the target speech. Based on their hypothesis that it is the accurate rendering of transients that is important for intelligibility, and the accurate rendition of steady voiced segments that is chiefly responsible for voice personality, they performed the transformation entirely in these latter segments, simply copying the other segments from the source speaker. They found that by adjusting the pitch and the spectral expansion they could produce many of the characteristics of the target speaker, but that non-linear spectral expansion, different spectral expansions in different segments, and altering the shape of a glottal pulse produced by a parameterised model all improved voice quality.

It is difficult to evaluate this work as a practical technology, since it was aimed at discovering what factors in a voice were essential to its personality. What it does demonstrate is that one needs a flexible spectral transformation, rather than a fixed normalisation, to achieve good quality, and that ultimately, voice transformations — and speaker modelling systems that support them — are going to have to account for speaker differences in the excitation signal.

Codebook based transformations

One of the earliest papers describing a voice transformation technology [shikano86], was, in fact, directed not towards producing speech matching a particular speaker, but towards frame-wise normalisation of speech from multiple speakers, allowing it to be used as input to a speaker dependent speech recognition system. Since this system formed the basis for the largest coherent body of work in voice transformation [abe88, abe89, abe91a, abe91b, mizuno94], it is worth discussing in some detail, omitting details of the normalisation scheme that did not survive in later incarnations.

In this system, the speech from the target speaker and the source speech were LPC encoded, and the LPC spectral coefficients vector quantised[1], separately for each speaker. The speech was then aligned by Dynamic Time Warping alignment [nye84], and, for each source codebook entry, a histogram was made of target codebook entries that were aligned with it. These histograms were used to create new entries, one for each entry in the source speaker's codebook, that were linear combinations of the target codebook templates, weighted by the histogram counts. This new codebook was used to quantise the speech, and the process was iterated, several times. Each source codebook entry then had a corresponding "mapping" codebook entry that was simply an average of the speech frames that were aligned with that frame.

In the case of the recognition system they were investigating, a single target speaker was used, and mapping codebooks were used to move the speech of several source speakers

1. Vector quantising a signal involves choosing a fixed-size set of templates into which a signal can be decomposed. The signal can then be efficiently transmitted by sending these templates once, followed by a sequence of template indices from which it can be reconstructed. Vector quantisation is widely used in speech compression, and, more relevantly here, in HMM based speech recognisers.

towards the target, resulting in an improvement from 64.0% to 83.1% in recognition accuracy.[2]

The development of this system for voice transformation will be discussed in the following paragraphs, but it is worth pointing out at this point that the quality of the transformation that can be produced is highly dependent on how good an alignment can be achieved between the source and the target speech. Imperfect alignments will result in mapping codebook entries that are smeared over many different target frames, resulting in poor quality synthetic speech. Most, if not all, of the papers discussed here have chosen or selected data in such a way that good alignments are easier to achieve[3], a luxury of choice that this author would have liked to have had available, considering the efforts that had to be exerted towards improving the alignments used in this synthesis task.

After their initial success with voice transformation for recognition, Abe *et al.* applied the VQ transformation technique to speech synthesis, with the eventual goal, shared by the work in this thesis, of retaining voice personality during speech-to-speech translation [abe88]. In this case, the VQ codebooks were supplemented by scalar quantised codebooks for pitch and power. The training samples were one hundred Japanese words per professional speaker used. Each word was uttered in isolation, again, making alignment reasonably straightforward. Voice transformation was achieved by simply applying the three mapping codebooks for the speaker pair in question and resynthesising.

The performance was measured both objectively and subjectively. The objective measure used was the difference in vector distortion between the transformed source speech and the target, compared with the original source speech and the target, and in the absolute size of the average pitch difference between the transformed pitch and the target. On the first measure, the transformation reduced distortion by 27% for transformations between two female speakers, of 49% between two male speakers, and by 66% between a male and a female speaker. The transformation reduced the average pitch disparity to less than 15Hz. Extensive use will be made of distortion measures like this in reporting the results of this chapter.

The subjective measures used involved presentation of the original and transformed speech to human subjects who were asked to make similarity judgements. Two experimental setups were used. In the first, the original speech, the target speech, and three versions of converted speech — one of which was produced missing the pitch and one missing spectral conversion — for a male-female speaker pair were presented in a set of forty randomly assigned pairs. The listeners were asked to make similarity judgements on a scale of "similar", "slightly similar", "difficult to decide", "slightly dissimilar' and "dissimilar". A multidimensional scaling technique [hayashi85] was used to display the similarity relations. This technique suggested that not only did the full conversion move the male speaker perceptually towards the female speaker, but that pitch and spectral alterations both made an important, independent contribution towards the perceived similarity of voices.

The second subjective measure was used to assess conversion between two male speakers. Words A and B from the two speakers M and N respectively were played, followed by a word X resulting from an M->N or N->M conversion. Listeners were asked to judge which

2. While we shall not dwell overly on matters we left behind in a previous chapter, it should be noted that the recogniser was trained in Speaker Dependent mode and used in Multiple speaker mode.

3. The Shikano et al paper [shikano86] for example, rejects training sentences for which the alignment cost is high. Rejecting outliers this way is an excellent practice, but is only possible if one has more data than one needs from each speaker.

token A or B the voice in token X most closely resembled. The best of the voice conversion efforts explored later in this chapter was assessed using this second listening test.

The main purpose in reviewing this work in VQ transformation was to introduce the problems of alignment and to explain techniques for assessing the quality of voice conversions. For the sake of completeness, however, and to preview some of the techniques that are applied here in different contexts, it is worth mentioning more recent versions of the system.

Since the system was intended for use in a speech-to-speech translation system, an attempt was made to determine whether there were consistent differences between speech in two languages that should be modelled during conversion [abe90, abe91a]. Codebooks were created for speech from a single, bilingual speaker speaking Japanese and English, and both codebooks were used to encode speech from both languages. Although differences were found, they were insignificant compared to interspeaker differences. It should be noted, however, that although the speaker was, apparently, at "native speaker" proficiency in both languages, there is a possible confound stemming from the use of a single speaker. Bilingual speakers tend to move their pronunciation of both languages towards an intermediate phonology [flege94]. Differences between languages may therefore be greater than those measured.

In the same papers [abe90, abe91a], Abe *et al.* introduced an application of voice transformation to speech synthesis, by applying it to the output of the MITalk [allen87]. speech synthesiser. This is similar to the technique adopted in this thesis. The synthesiser produced isolated Japanese words matching those uttered by a Japanese speaker. The phoneme string to be uttered was chosen by hand. MITalk synthesised it using American English phonological rules, except for duration rules, which were modified to more closely match those for Japanese. Mapping codebooks were derived from alignments of the MITalk "Japanese" words with the human speech, and used on the output of MITalk speaking American English, to produce English in the Japanese speaker's voice. Although the authors expressed some reservations about the quality of the speech produced, it is clear that applying target-speaker based transformations to synthetic speech is a useful path to take, if voice conversion is to be of significant technological import.

In [abe91b] this work was taken further. Instead of converting the speech frame by frame, whole segments were transformed. In this manner, it was possible for the system to convert both the static spectral qualities of the speech and the within-segment dynamics. This technique reduced the spectral distortion between the converted and target speech for a pair of males to one third of the distance between the original voices, and improved speaker ID accuracy by 20% over that of a frame by frame conversion.

A final VQ based paper [mizuno94] is worth discussing because it suggests why applying a universal function approximator, such as a neural net, to the problem of voice conversion, is a good idea. While this is not what Mizuno *et al.* did, their use of piecewise linear models of voice quality is a step in this direction. In this voice conversion system, each codebook entry for the source speaker's voice was analysed for formant frequency and spectral tilt. A linear model converting these values for members of the codebook entry into those of the target speaker was derived. During conversion, this model was used to specify formant frequencies and spectral tilt for the target speaker, and minimal spectral distortion (MSD) search was used to find the nearest matching frame for the target speaker. Target speech was

then resynthesised from this frame. Although the speaker ID accuracy for the resulting speech was actually slightly less than for VQ converted speech, perceived naturalness of the voice was higher. This suggests that smooth spectral transitions, of the sort that functional approximators can generate, are important for natural speech. If a neural network can be trained to produce the target spectrum with adequate fidelity, its smooth interpolation between frames should result in speech that is more natural than that from a system using VQ coding.

Neural net methods

In fact, some recent work has sought to take advantage of this power, at least with respect to the source speech. Nam and Savic [nam90][savic91] constructed a system that used a neural network *classifier* to select a target codebook entry given a frame of LPC coefficients of the source speaker. The training data was generated by doing a forced alignment, assisted by using voicing decisions, in the LPC cepstrum parameter space. During synthesis, the LPC coefficients from the target codebook were used either with a pulse train or with a frequency shifted version of the source excitation signal to produce the transformed speech. The authors reported that the modified source excitation signal produced higher quality speech.

It is not clear why the authors used the network as a classifier, rather than as a function approximator, although one might imagine that it was in an effort to reduce smear in the output LPC coefficients[4]. Besides this difference, the system that will be developed in this chapter could be seen as an extension of Nam and Savik's system to the use of synthetic source speech and plurispeaker synthesis using speaker model input.

A more sophisticated synthesis scheme

Valbret *et al.* [valbret92b] propose a scheme that treats the two main contributors to segmental variation, the voicing source and the spectral characteristics, separately. Their work was also motivated by the notion that a "speaker can be characterized by a 'spectral print' in some parameter space", although they do not try to construct such a model explicitly. In their review of the work of Abe *et al.* and Nam *et al.*, they claim that the voice quality in those systems was limited by the use of the LPC vocoder. In their system, the authors decomposed the input signal into the excitation signal, and a series of pitch synchronous LPC spectral envelopes. The spectral envelopes were divided into disjoint acoustic classes and either a dynamic frequency warping (DFW) or linear multivariate regression (LMR) based transformation was learned between the source and target vectors in these classes. The target speakers's excitation signal was used together with the transformed spectral envelopes were used to produce a target waveform for each pitch period, and the PSOLA synthesis technique was applied to combine these waveforms into the target speech signal. Because prosodic modelling was considered to be beyond the scope of the paper, the actual pitch and timing of the target signal were applied to the transformed speech[5].

4. Although this is a reasonable hypothesis, a preliminary attempt to do the same in the voice transformation system developed here produced no improvement in the synthesised speech compared to direct transformation. Since time was limited and a detailed comparison of voice transformation systems was somewhat peripheral to the problem of model based synthesis, this attempt is not reported in detail here.

Training was performed using specially recorded CVC logatoms, which were used because the authors were unable to obtain sufficiently accurate time alignments for training speech consisting of simple noun phrases[6]. This difficulty is mentioned here because it is likely that alignment problems were partially responsible for the comparatively low quality of the transformed speech produced in the experiments reported in this thesis.

Of the two techniques, DFW and LMR, the authors reported that LMR produced speech that more closely matched the voice of the target speaker. This is promising for the neural network techniques used here, since, as has been pointed out in great detail earlier, the use of a neural network as a functional approximator is exactly a non-linear multivariate regression and often degenerates to being exactly an LMR model. In the current system, however, instead of dividing the acoustic space into non-linear regions and hoping for piecewise linearity within those regions, the system relies on the possibility of non-linear mapping by the network to build a unified transformation of the entire acoustic space.

6.2. Introduction to the Experimental work

The remainder of this chapter is a description of experiments in which a voice transformation system with multiple target speakers was developed, modified to use the SVCs described in chapter 4, and then evaluated by human listeners.

First, though, Section 6.3 briefly describes an attempt to produce the target speech directly from the speaker models, rather than by transforming another voice. Since this attempt was unsuccessful, the system was built by producing a neural network function approximator that converted the output of a commercial text-to-speech system into the target voice. This allowed the transformation network to concentrate on altering only the spectral characteristics of the speech, about which the voice code could be expected to contain useful information.

The voice transformation network was rather similar to the speaker dependent recognition networks described in the previous chapter, but trained to produce an LPC frame in the target voice instead of a phoneme label. Frames output by this network could be synthesised by the LPC10 vocoder. Details of the network, and of the alignment between the input and the target speech, are given in Section 6.5 to 6.7. Initial evaluation of the transformation network (Section 6.8) showed that, while the quality of the speech produced was poor, the neural network was able to move both the pitch and the filter characteristics of the input speech closer to those of a single target speaker (Section 6.8.2). In the experiment described in Section 6.8.3, the transformation network was given *1-from-n* speaker IDs, and was able to move the speech closer to that of any chosen target speaker from a set of five.

Section 6.8.4 describes an initial attempt to drive the transformation from a speaker code. For this experiment, the code was generated by averaging eight-element PCA-based phone codes for each speaker, and then compressing the concatenated average phone codes in a bottleneck network. This yielded one four element speaker code per speaker. Comparison of the output of a transformation network trained to use this code with the natural speech of a

5. Given the importance that pitch and timing seems to play in determining speaker identity, one would expect this to improve the perception of target speaker identity in the transformed speech markedly.
6. These noun phrases consisted solely of an article, an adjective and a noun.

set of speakers, including the target speaker, showed that the network produced speech that was, on average closer to that of the target speaker than to that of other speakers.

Finally, a voice transformation network was built that was designed to use the actual speaker codes from one of the speaker models described in Chapter 4. This, network, whose construction is detailed in Section 6.9., could produce speech intended to mimic that of any speaker in the TIMIT speaker set. Testing, again by measuring the best alignment distance between the output speech and natural speech from a large group of speakers, including the target speaker, showed that the transformation was moving the speech towards the specified target on average. This system produced the best quality output speech, presumably because the amount of training speech was much larger than for the other transformations, but the quality was still poor. Because voice transformation *per se* was not the goal of the thesis, it was not possible to invest the work that would have been required to produce high quality speech from the basic transformation.

After the model-based transformation system had been built, a final series of experiments was run to see whether human listeners could identify the output speech with the voice of one of a pair of speakers. ABX designs were used in all cases. In the first of these experiments, described in detail in Section 6.11, speech was produced at every stage of the conversion process between natural and DECtalk speech to asses how each step affected voice personality. Even LPC coding the natural speech and time aligning it with the synthetic speech significantly affected the perception of voice personality. For the speech output by the transformation network, the target speaker could be identified only in the case where the other speaker choice was not the same gender. This finding was consistent with the earlier indication that most of the useful information in the task independent speaker models concerned speaker gender. What remained to be seen was whether this distinction was supported entirely by the network's ability to change the pitch of the DECtalk speech. In Section 6.11, the subjects compared the output of the transformation with DECtalk speech whose pitch contour had been replaced with that of the target speaker or a different speaker. Subjects were able to distinguish the speakers when their sex differed, using no other information than pitch. In Section 6.13, an attempt was made to see whether, given that pitch was sufficient to account for the amount of voice personality in the transformed speech, it was also necessary. The output of the speech transformation, leaving pitch unconverted, was compared with human speech that had been time aligned with the DECtalk utterance, and that had had its pitch contour replaced with that from DECtalk. Deprived of pitch information, subjects were unable to identify the correct speaker, even when the speaker gender differed.

Although the voice transformation achieved was not of high quality, and served mainly to confirm the difficulty of forming task-independent speakers models of general utility, it is likely that it could be improved. With sufficient engineering work, some suggestions for which are given in the conclusions for this chapter, it seems likely that a plurispeaker voice transformation system could be built with quality comparable to voice transformation systems reported in the literature.

The following sections describe experiments in more detail. A reader wishing to gain only an overview may wish to skip forward to the conclusions in Section 6.14 on page 166.

6.3. Initial work - direct synthesis from models

Since the aim of working with synthetic speech was primarily to provide a test-bed for the speaker modelling work, some initial experiments were done using the rather radical approach of trying to synthesise speech directly from a speaker model. The speaker models that are built using compression are invertible, in the sense that they can be used to estimate the input that produced the low dimensional representation representing the speaker's voice. If they are based on a representation, such as LPC coding, allowing convenient resynthesis, they can be used "in reverse" to produce speech. Although these experiments were done using earlier, pilot versions of the speaker models, and might be improved by the use of the final speaker model, neither of these methods produced even marginally acceptable speech quality, and they are chiefly included here to as an explanation for the move to voice transformation. Two methods for producing speech directly from speaker models were tried:

6.3.1. Inverting the model all the way through

First, the overall speaker model was expanded to produce estimates of each of the phone codes. Then the phone codes were each expanded to produce fixed length vectors in the same vector space as the original speech. These vectors were simply concatenated to produce the desired phone sequence, and the result resynthesised. The resulting incoherent babble discouraged pursuit of this method of synthesis.

6.3.2. Concatenating segments

Given the recent success of systems doing concatenative synthesis [sorin91, hauptmann94], it seemed reasonable to expect that the speech quality could be improved if, instead of using the inverted model speech directly, this speech was replaced by that section of actual speech from the database most closely matching the speech produced from the model. The speech was additionally constrained to match the phonetic context desired, so long as sufficiently many choices were available with that context. For this experiment, the PSOLA technique, which has enabled much of the recent success of concatenative synthesis, was not applied. It didn't seem warranted for the initial experiment, and as it turned out, the speech produced by segment concatenation was not noticeably better than that produced by direct model inversion. It is unlikely that PSOLA would have improved it significantly.

6.4. Voice transformation.

It had become clear at this point that, even if the speech produced by direct model inversion was matching speaker voice characteristics at a segmental level, the underlying *synthesis* technology of naive concatenation would never produce a convincing demonstration that speaker modelling was a useful technology for practical purposes. However, as mentioned in the introduction to this chapter, other researchers had claimed good performance from systems that transformed the voice of one speaker into that of another. It is clear why doing this transformation should be relatively straightforward, compared to the task of synthesising a new voice whole. One is not forced to reproduce every characteristic of the speech — the

information bearing properties, including the sequence and articulation of phones, and the pitch and amplitude contour can be taken from the original voice and altered as necessary. They do not need to be created anew starting only with a string of phone labels, they only have to be augmented with the voice characteristics of the target speaker. Abe et al [abe91a] used similar reasoning, pointing out that in attempting to produce English speech in a Japanese voice, it was necessary to retain dynamic characteristics of MITalk speech, which help the speech sound like English, while changing the spectrum into that of the target speaker, since static spectral characteristics provide clues to speaker identity.

Although the goal of producing any modelled speaker's voice as the target was somewhat more ambitious than the usual single speaker pairs used in the literature, the underlying ideas that had been developed to support such transformations were easy to apply.

6.5. Transformation Method

The general technique used was relatively straightforward, and amounted to a non-discrete version of Savic and Nam's [nam90, savic91] codebook based voice transformation network which was described above, with, of course, additional inputs for the speaker model.

A connectionist network was used to perform a non-linear transformation from a combination of a number of LPC frames of input speech from the source speaker and the speaker model representing the target speaker, into a frame of speech in something approximating the target voice. An example of such a network is shown in Figure 38.

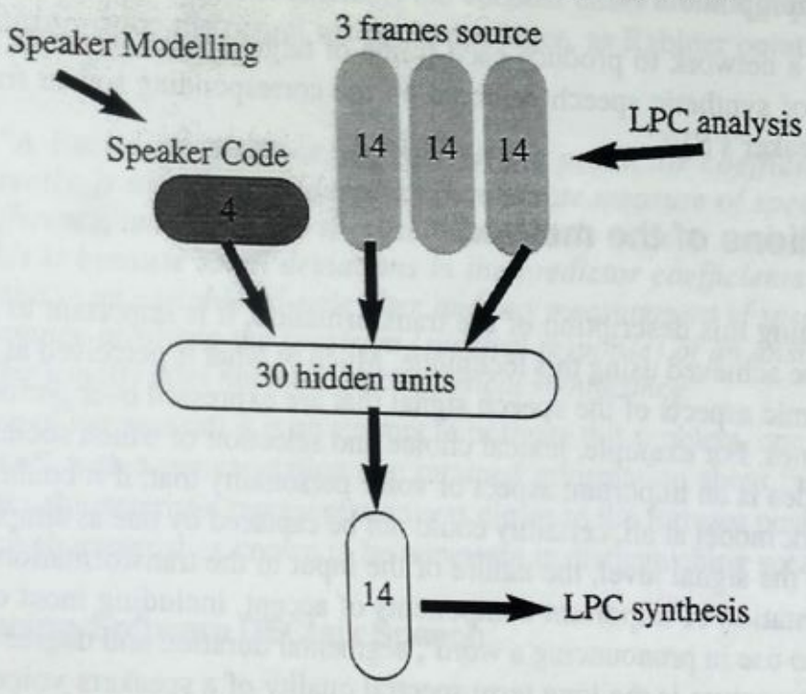

Figure 38: The voice transformation network combines source speech with a target speaker model to produce target speech. One example network is shown, but variations with different numbers of input frames, speaker code widths and number and size of hidden layers were also used.

Obviously, since the speaker modelling experiments were done with TIMIT speakers, it was desirable to use these speakers as the target set of voices for synthesis. To train a voice transformation, though, one needs to have available corresponding speech from two different speakers. Since it was intended that the system should achieve transformations to multiple target speakers, it was necessary to have a single source speaker produce all the training utterances for all the target speakers. The author's voice, since its pronunciation of words differs dramatically from the TIMIT speakers', could not be used. Even if its use had been possible, uttering of the order of a thousand sentences for this purpose would have been rather taxing. Fortunately, just as this section of the work was beginning, Digital Equipment Corp. (DEC) released Software DECtalk [dectalk94] a software-only version of the well known speech synthesis product. Software DECtalk provides a consistent source voice that will say more or less whatever one pleases. Moreover, since the program can be persuaded to output phoneme boundary information as it speaks, the experimenter is freed from the need to label the source speech by hand.

Training a voice transformation was done in four stages, which will be described in more detail in following sections:

- Producing Software DECtalk Speech for utterances corresponding to the training speech available for each TIMIT speaker.
- Time aligning the target speech with the corresponding Software DECtalk speech.
- Locating the target speaker in speaker space by using the speaker models to generate the appropriate SVC.
- Training a network to produce each frame of target speech, given as input both a window of synthetic speech centered on the corresponding source frame and the target speaker's SVC.

6.6. Limitations of the method

Before continuing this description of the transformation, it is important to recognise what can and can't be achieved using this technique. Much of what is perceived as voice quality is based on dynamic aspects of the speech signal that are expressed over periods much longer than a few frames. For example, lexical choice and selection of which social register to use to express an idea is an important aspect of voice personality that, if it could be captured by a neural network model at all, certainly could not be captured by one as simple as those used here. Closer to the signal level, the nature of the input to the transformation used here precludes representation of important components of accent, including most of the choice in which phones to use in pronouncing a word[7], segmental duration and degree of stress. What one can hope to capture is the long term spectral quality of a speakers voice, and, perhaps, specific qualities of spectral transitions that occupy only a few frames.

Although these omissions impose significant constraints on the system's ability to achieve convincing mimicry, and it would certainly be desirable to address them in future work, they are limitations that are shared by all the other voice personality transformation systems

7. Although if a speaker made utterly consistent substitutions, the network might be able to capture some of them.

reviewed. Although one hopes that these longer term effects can eventually be addressed by voice transformation systems, it is currently worth pursuing the goal of obtaining good transformations of the short term components of voice personality.

6.7. Details of the transformation

6.7.1. The speech representation

In all these experiments, the speech was encoded using the standard LPC10 encoder [tremain82], with the quantisation component defeated, and the LPC reflection coefficients converted into the Log Area Ratios described in earlier chapters [rabiner93, Bridle 1994 personal communication][8]. The encoder was also modified to eliminate a three frame delay that had prevented direct use of the TIMIT label files with the LPC10 encoded speech. The speech representation consisted, therefore, of a series of 22.5ms non-overlapping frames each containing fourteen values: two half-frame voicing decisions, pitch, power, and ten LAR LPC coefficients.

No claims are made here that the LPC10 coder is the best, or even a good, choice for doing this work. It was simply available, and easy to modify. Future versions of the system would almost certainly be improved by choosing a better parameterisation of the speech signal. Besides the distortion inherent in a low frame rate coder with pitch and half frame voiced/unvoiced decisions as its only source of voicing information, there is the additional difficulty that the LPC reflection coefficients the vocoder emits are not good candidates for the kind of transformation the neural net performs, since, as Rabiner points out [rabiner93 p191]:

> *"A Euclidean distance, defined on the predictor coefficients directly, is usually considered an inadequate measure of spectral difference, unless the two spectra are extremely close to each other. This is because small deviations in the predictor coefficients can result in an unstable all-pole filter, and any measurement of spectral distance involving the spectrum (spectral response) of an unstable filter usually does not have much physical significance"*

While the use of log area ratios is an attempt to mitigate this problem, one would undoubtedly be better off with a representation that retained information about the glottal source, and whose filter characteristic representation was closer to the formant positions and width, and other voice characteristics known to be important in distinguishing speakers.

6.7.2. Producing Software DECtalk Speech

The TIMIT database used contained both word level and a phoneme level transcriptions of the speech. If a human source speaker had been used, they would have been instructed to produce an utterance matching the word level transcription. With Software DECtalk one can do slightly better than this; it can be instructed to produce a phoneme sequence nearly iden-

8. Log Area ratios can be calculated from LPC reflection coefficients. They are intended to approximate the log ratio of the areas of successive equally spaced approximately cylindrical sections through the vocal tract.

tical to that used by the original speaker. Since the phoneme labels used in the TIMIT transcriptions, and those required by Software DECtalk differ somewhat, a rather *ad hoc* conversion was done, specified in Appendix A.3.

The Software DECtalk software was used to convert each phoneme string into a digital audio file of the speech and to produce a corresponding phonetic transcription[9]. Each speech file was encoded, using the LPC10 encoder, in the same way as the target speech, and the timing information in the transcription was converted so that timings were given in terms of LPC frame indices.

6.7.3. Aligning the speech

In order to learn a mapping between the source and target speakers' speech, it was necessary to find a correspondence between the Software DECtalk speech generated for each utterance and each training speaker, and the natural speech that was the target. It has been noted, above, that doing this well is not a trivial affair when two human speakers are involved, and, unfortunately, it has been shown [hunt84] that aligning MITalk[10] synthesised speech with human speech, approximating what is being done here, is much harder yet. Some effort had to be invested in the alignment process, and even so, the alignments produced were not always as accurate as one might have wished.

The TIMIT speech was aligned with the Software DECtalk speech by subtracting overall means from the frames in each file before alignment and then finding an path minimising the total frame distance $d = (0.1v_1)^2 + (0.1v_2)^2 + (0.005p)^2 + 0.01e^2 + \sum c_i^2$, where v_1, v_2 were the differences between the two frame voicing decisions, p was the pitch difference between the two frames, e was the RMS energy difference, and c_i was the difference between the i^{th} LAR coefficients. An additional constraint was applied to ensure that a single frame of speech took up no more than five frames after warping. The alignment was set up so that the target speech was distorted to match the timing of the source speech, enabling arbitrary source speech to be used during testing.

Initial experiments with this, whole sentence, alignment suggested that the alignment process did not produce very good speech. Plotting just the energy contour for the original, dectalk, and aligned speech made it clear that simply minimising a simple frame distance did not produce the precise alignment needed to learn the best possible mapping between source and target speech. In fact, it did not come close.

The reason for this was not clear, but the choice of spectral representation seemed to be a possibility. Although Rabiner and Juang [rabiner93, p191-2] suggested that the LAR measure used ought to have been a reasonable choice for use with a Euclidean distance metric, it seemed prudent to try using the more popular cepstral distance. Although a definitive set of experiments was not done, comparing the alignments produced by the two measures suggested that, if anything, the LAR measure was performing better[11].

---

9. This transcription did not always correspond exactly to the phoneme sequence that was input. It appears that Software DECtalk contains some obligatory phonological rules.
10. Or, in this case, Software DECtalk.
11. It should be emphasised that much more testing and parameter tweaking was done for the LAR measure, so little should be read into this, except that the LAR measure appeared to be a reasonable one to use.

In order to improve the alignment by using more information, the labelling information available for the two speech samples was used to identify areas of correspondence. A similar alignment procedure to that described above was used, but now the path could be constrained so that frames known to correspond in fact, could be forced to coincide in the alignment. At first, the intention was to align the starts of words within the sentences, but, promises in the manual notwithstanding, Software DECtalk could not be persuaded to output whole word timing information. Instead, it was necessary to use phoneme based constraints. While this produced a more constrained, and, one would hope, more accurate alignment, it was more difficult to achieve, since the phonemes and phoneme labels used by the TIMIT database labelling and those produced by Software DECtalk do not exactly coincide.

For this reason, alignment paths for the LPC coded speech were forced to coincide at a subset of phoneme starts, themselves chosen by a lexical alignment of the phoneme labels for the two strings. The distance between phoneme labels was set to zero if they were identical, to 0.5 if they shared a first letter[12], and to one otherwise. Only exact coincidences (zero distance) were used to constrain the path, and then, only if they were separated by more than one frame. Figure 39 shows the energy contour of speech aligned using this phoneme constrained alignment, compared with the original speech and the Software DECtalk speech for an example sentence from the database.

6.8. Testing the basic voice transformation method

Although the neural net converter chosen was not entirely dissimilar to those reported in the literature, especially in [abe91b, mizuno94 and nam90], it was important to establish single speaker performance as a baseline against which multi-speaker and model-driven voice conversion could be measured. The first goal was to establish whether Software DECtalk speech could be transformed into something more closely resembling the voice of a single human speaker.

6.8.1. Evaluation technique

After the networks had been trained, the quality of the transformation could be evaluated by simply measuring the average frame distortion (d_c)[13] for a DTW alignment between the transformed synthetic speech, and the aligned target speech for the same sentence. The smaller this distortion, the better the transformation achieved. This distortion was evaluated by comparing it with the original average frame distortion (d_o) between the unconverted synthetic speech and the target; any decrease indicated that the transformation had achieved some success. It was also useful to express the results as a percentage distortion decrease $100 \times (d_o - d_c) \div d_o$ relative to the unconverted speech distortion.

To compare the extent to which changes in pitch — the most obvious component of voice quality — as opposed to changes in the spectral shape represented by the ten LAR coefficients in the frames, contributed to the transformation achieved, percentage distortion reduc-

12. This is an odd criterion, but worked well for the TIMIT and dectalk-output phoneme labels.
13. Defined in section 6.7.3 on page 146.

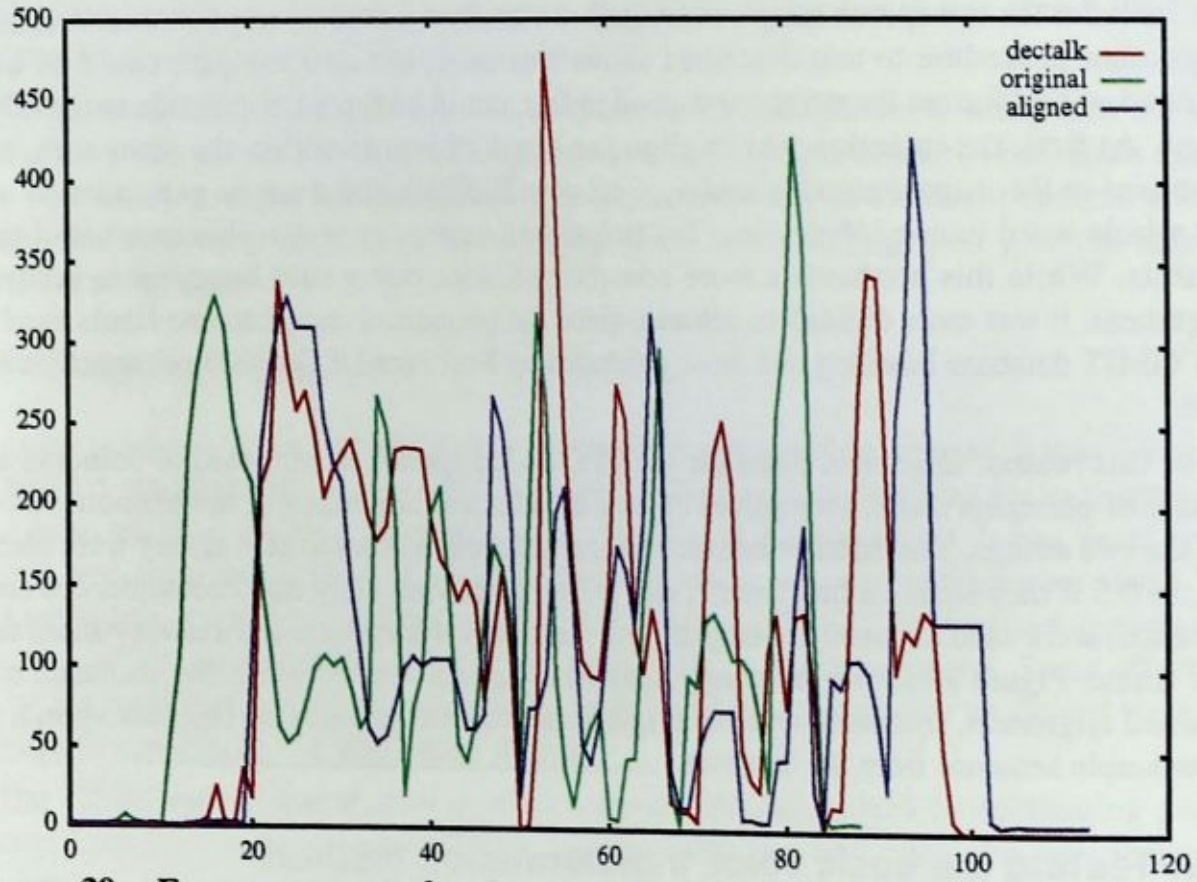

Figure 39: Energy contours for the original speech, the corresponding Software DECtalk speech, and the original speech aligned to the Software DECtalk speech using the techniques described in the text. The speech in question is the TIMIT sentence sx75 — *"The prowler wore a ski mask for disguise."* — spoken by training set speaker dr1-mgr10. Although the alignment improves the correspondence between the energy contours as it is designed to do, it is evident that there is still a considerable difference between the two speech signals and a great deal of opportunity for misalignment.

tions were also measured for alignments using *only* the pitch and spectral components respectively.

6.8.2. Experiment: Transforming Dectalk speech into single Human speaker

Table 41 gives these distortion measures for a variety of neural network architectures applied to the task of converting speech from Software DECtalk into the voice of the TIMIT speaker *fsjw0* from dialect region three. The performance figures are for the TIMIT sentence sa_1, which was not used in training the conversion. Training parameters common to the networks are given in Table B-1 on page 182 of the Appendices.

The average frame distance for an alignment between the target speech and the Software DECtalk input is given at the top of the table for purposes of comparison, together with the distances for an alignment using just the ten LPC-LAR coefficients, and an alignment using just pitch.[14] The remaining rows of the table give the corresponding distances after the Soft-

14. In this case, the distances given are the average difference between the unweighted pitch values from the aligned frames.

ware DECtalk speech had been converted using neural networks with between one and thirteen frames of input, and between thirty and one hundred hidden units, arranged in one or two hidden layers.

The closest match with the target speech was attained by the simplest network, which converted the speech in a single source frame at a time using only thirty hidden units.

Table 41: Effects of voice conversion of Software DECtalk speech with a single human speaker as the target. Distances of converted speech from the target are given for a variety of neural network architectures.

		Distance from Target Speech			% Improvement by Conversion		
		Whole Frames	Spectral	Pitch	Total	Spectral	Pitch
Raw Software DECtalk		3.24	1.44	24.2	0	0	0
Input Frames	Hidden Units	Converted (d_c)	Spectral	Pitch	Total	Spectral	Pitch
1	30	1.99	0.85	8.51	38.6%	41.0%	64.8%
5	30	2.18	1.09	9.03	32.7%	24.3%	62.7%
	60	2.14	1.06	8.98	34.0%	26.4%	62.9%
9	30	2.06	1.09	9.10	36.4%	24.3%	62.4%
13	30	3.39	1.94	8.30	-4.6%	-34.6%	65.7%
	60	2.55	1.12	10.0	21.3%	22.2%	58.8%
	30-30	4.11	2.36	9.95	-26.9%	-63.9%	58.8%
	50-50	3.70	2.18	11.8	-14.2%	-51.4%	51.2%

The networks with the widest input windows actually produced speech that differed from the target speech to a greater degree than the unaltered speech from Software DECtalk. Since there were only a total of nine training utterances containing 1 305 frames of speech available for this speaker, it is reasonable to suspect that the main reason for the superiour performance of the small network on testing data was that the larger networks were overfitting the training data. This hypothesis is supported by the observation that pitch conversion was moderately successful for all the networks, since even a small amount of training

speech ought to be enough to estimate reasonable linear model of the pitch change. Further

Table 42: Training and testing set errors for single speaker conversion networks. These errors are the usual Euclidean distance used with connectionist training.

Input Frames	1	5		9	13			
Hidden units	30	30	60	30	30	60	30-30	50-50
Mean Train Error	0.506	0.503	0.522	0.389	0.143	0.437	0.059	0.073
Mean Test Error (sa_1)	0.505	0.537	0.527	0.702	1.66	0.71	1.811	1.620

support is provided by the mean output unit errors measured during training, for the networks for training frames and for frames from sa_1, which are given in Table 42. The networks with more than five input frames showed clear signs of overfitting, although this noes not seem to have greatly harmed performance in the case of the network with nine frames of input. The differences between the performance of the single frame network and those with five frames may be due to over-fitting, or may simply be due to chance differences in the nature of the conversion function learned from the training data.

Conclusion

The effects of voice conversion were not just confined to a pitch normalisation, although such a normalisation was performed by the network. In almost all cases, the conversion network moved the spectral representation of the synthetic speech, contained in the ten LAR coefficients, substantially closer to that of the target speech.

Although the speech produced was not of high quality, this experiment had verified that a single neural net, acting as a function approximator, could successfully transform the speech of the Software DECtalk synthesiser into something more closely matching the speech of a target speaker.

6.8.3. Experiment: Plurispeaker synthesis, using perfect speaker information.

If the transformation was to be driven by the speaker model, as intended, the transformation network would have to produce speech from more than one speaker. Following the pattern of the recognition experiments, a *one-from-N* representation was used as an idealised speaker model.

To this end, separate transformation networks were trained for speech from five individual speakers. The task was to convert a single frame of input speech into the corresponding target speech, using the additional information supplied on five speaker ID units. Three varieties of networks were used, the shared parameters for which are given in Table C-2 in Appendix C. The first was a linear network, with no hidden units, that could compute only linear transformations of the input units - this network was capable of only a simple spectral, power, and pitch normalisation, and was included to act as a baseline for comparison of the performance of the non-linear networks with greater powers of functional approximation [hertz91]. The other two networks both had thirty hidden units. In one network, the hidden

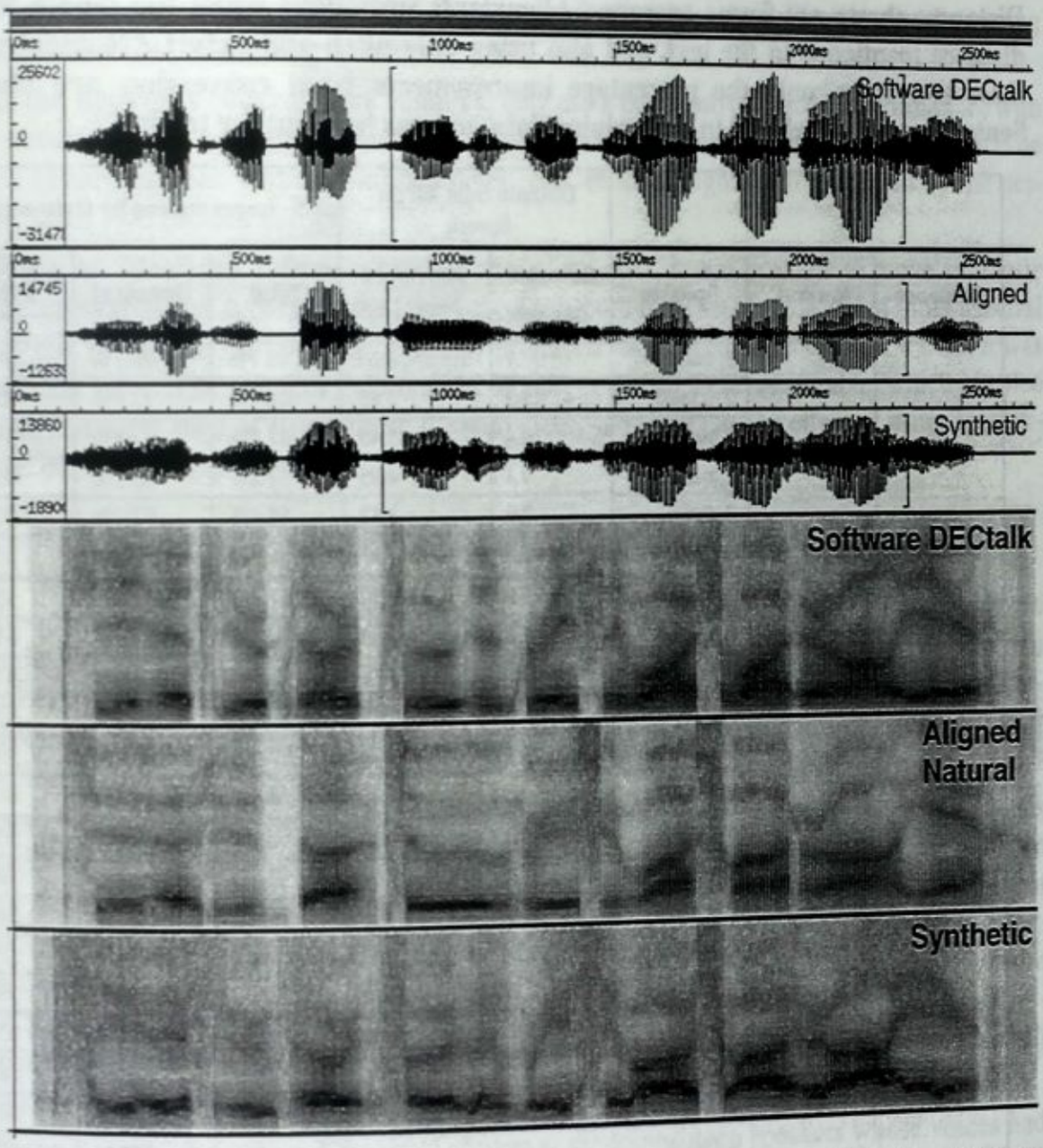

Figure 40: Software DECtalk speech, natural speech aligned to the Software DECtalk speech, and synthetic speech produced by the transform network, both as waveforms and as spectrograms. The transformation network used was one using a *1-from-n* input encoding to select among a set of training speakers. The synthetic (converted) speech seems to share qualities of both the Software DECtalk and target speech, but shows substantially more spectral smear than either. This smearing is a likely contributor to the poor quality of the synthetic speech after transformation.

units were arranged in a single layer, and in the other they were arranged in two layers of fifteen units each.

The following table (Table 43) gives distortion measurements for the three types of networks, both for a sentence included in the training data (sa_2) and one that was held out for testing (sa_1). With the exception only of the linear network applied to test set speech for two speakers and training speech for one speaker, the transformation networks altered the synthetic speech so that it more closely matched speech from the target speaker. As one would

Table 43: Speaker dependent Voice Transformation for five randomly chosen speakers. Distances shown are frame averages. Alignments were done using the whole frame distance mentioned in the text, and also using only pitch and only LAR spectra. In these two cases, only the percentage improvements from conversion are shown. Sentence sa_2 was included in the training data, sa_1 was held out for testing.

			Distance from Target Speech		% Improvement by Conversion		
Sentence	Network	Speaker	Software DECtalk	Converted (d_c)	Total	Spectral	Pitch
sa_1	Linear	3_fntb0	9.11	4.67	48.7%	69.1%	48.2%
		2_fcmm0	12.3	6.83	44.4%	93.9%	48.6%
		2_mmag0	10.4	5.45	47.8%	8.48%	39.6%
		1_mklw0	9.3	4.35	53.2%	15.1%	45.4%
		3_fsjw0	7.9	3.52	55.4%	82.7%	57.1%
		mean	9.81	4.97	61.3%	50.1%	55.5%
	TwoHid	3_fntb0	9.11	4.44	51.3%	67%	53.3%
		2_fcmm0	12.3	4.67	62%	80.9%	61.4%
		2_mmag0	10.4	4.76	54.5%	16.6%	47.1%
		1_mklw0	9.3	3.83	58.8%	22.9%	53.5%
		3_fsjw0	7.9	3.36	57.5%	78.4%	59.5%
		mean	9.81	4.21	70.8%	47.2%	64.4%
	OneHid	3_fntb0	9.11	4.28	53%	65.5%	53.7%
		2_fcmm0	12.3	4.97	59.6%	87.4%	58.5%
		2_mmag0	10.4	4.1	60.7%	2.32%	56.3%
		1_mklw0	9.3	3.8	59.1%	9.54%	53.4%
		3_fsjw0	7.9	3.36	57.5%	77.5%	59.9%
		mean	9.81	4.1	72.2%	46.2%	65.9%
sa_2	Linear	3_fntb0	10.7	3.05	71.6%	89.4%	68.8%
		2_fcmm0	10.6	4.53	57.1%	86.9%	58.9%
		2_mmag0	8.27	3.14	62%	1.23%	57.6%
		1_mklw0	9.1	3.54	61.1%	9.17%	54%
		3_fsjw0	8.28	3.01	63.7%	82%	62.7%
		mean	9.39	3.45	71.7%	51.9%	68%
	TwoHid	3_fntb0	10.7	2.79	74%	77.8%	71.6%
		2_fcmm0	10.6	3.59	66%	83.7%	68.8%
		2_mmag0	8.27	2.85	65.5%	-19%	62.3%
		1_mklw0	9.1	2.89	68.2%	-19.7%	62.5%
		3_fsjw0	8.28	2.79	66.4%	80.7%	65.2%
		mean	9.39	2.98	77.4%	46.1%	74.4%
	OneHid	3_fntb0	10.7	2.75	74.4%	81.7%	72.4%
		2_fcmm0	10.6	3.37	68%	84.6%	70%
		2_mmag0	8.27	2.8	66.2%	-0.537%	62.3%
		1_mklw0	9.1	2.82	69%	-3.33%	62.3%
		3_fsjw0	8.28	3.01	63.7%	77.3%	61.7%
		mean	9.39	2.95	77.8%	48.3%	74.2%

expect, the transformation matched the speech (sa_2) from the training set more closely than held-out speech. There was about a third less reduction in overall distortion for the testing speech than for the training speech.

As the networks became more complex, they did a better job of voice transformation, with the network with a single hidden layer substantially outperforming the linear network. The three layer network with one hidden layer, was, in turn, slightly outperformed by the network with two hidden layers.

Using an idealised *1-from-n* speaker model as input, it was possible to move synthetic speech towards the speech of the target speakers, both in terms of pitch and of other spectral features. While the output speech quality produced was inadequate for practical uses, it was clear that effects of speaker information on output speech were measurable, which enabled taking the next step of using those measurements to investigate the effect of using speaker-space based speaker models.

6.8.4. Experiment: Plurispeaker synthesis, mean speaker models.

Having established that neural networks could use speaker information to move synthetic speech towards that of a target speaker, the next goal was to determine whether positions in speaker space, automatically derived from speech, could be used to train the voice transformation, and whether, after that had been done, the transformation was useful for new speaker targets.

The speaker model used in this experiment was generated by first linearly warping phones to ten fourteen-element frames each, and then projecting these 140-element vectors onto their first eight principal components, computed within each phone. These eight element phone codes were averaged for each speaker, and concatenated into a 488 element vector. The one hundred and eighty-nine such vectors corresponding to the training speakers were then used to train a neural network compressor[15]. This network was used to produce four-element speaker codes for all one hundred and eighty-nine training and sixty-two testing speakers.

Three sets of these speakers were chosen at random: fifteen speakers whose voices had been used to train both the speaker model and the transformation network (*training*), fifteen test speakers who had been used to train the speaker models but who were not used to train the voice transformation (*test*), and fifteen test speakers who had not been used previously (*true test*).

A voice transformation network was trained in a similar manner to that in the preceding experiment except that the four component mean speaker voice code for each speaker was used to replace the binary speaker ID used in the previous experiment, and that three frames of speech, centred around the target frame, were used as input to the transformation.[16] The transformation network had thirty hidden units in a single hidden layer, and additional bypass connections were present directly connecting the input and output layer.

---

15. The network had 488 inputs and outputs, and twenty units in the first, four in the second, and thirty in the third of three hidden layers. No bypass connections were used. Speaker codes were extracted from the four-unit bottleneck layer. The network was trained for 8 000 epochs with a learning rate of 0.001 and momentum of 0.8.
16. Details of the training parameters are given in Table C-3 on page 184 in the Appendices.

Testing the model based transformation

A difficulty with these voice transformation experiments is determining whether one has succeeded in producing the modelled voice. In recognition experiments, the matter is straightforward: if recognition accuracy is better with the speaker information than without it, then one can be said to have succeeded, although the details of how this success was achieved may be difficult to discern. In synthesis, no such simple scoring criterion is available. What must be done is to compare the synthetic speech with actual speech from the target speaker, and see whether it matches that speaker more closely than other speakers. Fortunately, this is possible for the TIMIT database, which provides two particular sentences which are said by every speaker. One of these, sa_1 (*"She had your dark suit in greasy wash-water all year"*) was selected for use in testing.

Within each group of fifteen speakers, synthetic sa_1 speech s_i was produced for eachspeaker. Natural speech, N_i was also available for each speaker. An average alignment distance D_{ij} could be computed between the synthetic speech from any speaker i and natural speech for any speaker j by measuring the average distortion of a frame on the best alignment path. These distortion measures form a 15×15 matrix, with the distance between natural and synthetic speech for each speaker lying along the diagonal, and the distances between the synthetic speech for a speaker, and natural speech from other speakers, off the diagonal. If the speaker model is successful, then the diagonal elements should be row minima. To measure whether this was the case, the diagonal elements were subtracted from each row, and the elements of the matrix summed. The result was then normalised by dividing by the number of elements in the matrix, in this case 225.

A positive value of this measure is an indicator of success. Table 44 gives this measure for each of the three speaker groups used in this experiment.

Table 44: The effect of speaker model input on the match between synthetic and natural speech is shown here for training speakers, speakers used to train the speaker model, but not the voice transformation, and completely untrained speakers. The measure, described in the text, compares the match between synthesised speech for a particular speaker and that speaker's natural speech with the match between the synthesised speech and natural speech from other speakers. Larger values represent better conversion.

Speaker Group	Measure
Train	0.397
Test	0.119
True Test	0.105

The speaker model clearly moved the synthesised speech in the direction of the target speaker, on average, with the strongest effect being seen on speakers within the training set. As one would expect, the effect of the speaker model was less pronounced for speakers who

had been used to train the speaker modelling network, but not used to train the voice transformation, and still less pronounced for speakers who had not been used in training at all.

Using a real speaker voice code generated by a network modelling speaker variation, it was possible train a voice transformation to convert synthetic speech into something, resembling, on the average, the target speaker more closely than other speakers, even for speakers who had never been encountered in training. Although a significance test for this effect was not readily available, there was evidence that speaker spaces could be used to model speaker variation in a way that the voice transformation could use, and in a way that generalised to new speakers.

6.9. Voice Transformation using a speaker model for the whole database

Having established that the simplified speaker models of the last two experiments could affect the voice transformation in an appropriate, if insufficiently accurate, way, the final step in the development of the system was to use one of the fully developed speaker models from Chapter 4 to build a transformation network that could cover the complete set of speakers in the TIMIT database.

6.9.1. A neural net speaker model used for voice transformation.

The speaker model used for this purpose was the NNCCR2 (Neural Network Compression with Pattern completion and Recirculation of outputs to inputs) model with fifteen speaker model units as described in Section 4.6.

During training, fifteen-dimensional speaker codes were extracted from this model after all the available speech had been presented to the speaker modelling network[17], and these codes were presented as speaker input to a transformation network with a five frame input window.[18] The training data for the transformation itself consisted of all of the available speech from the training set speakers, except for that occurring in the utterance of sa_1 by each speaker, time aligned with the same utterance as spoken by Software DECtalk. After the network was trained, using the parameters given in Table C-4 of Appendix C., its performance was tested using the same technique outlined in section .

The speaker modelling networks were used to generate speaker codes for each speaker in the training or testing set at the point when two hundred phones had been heard, or after all the speech from the speaker[19] had been exhausted, whichever came first. These speaker codes were used, along with Software DECtalk's utterance of sa_1, to produce synthetic speech in every modelled voice. This modelled speech was compared with the actual utterances of sa_1 by the speakers, after they had been time aligned with the Software DECtalk

17. Actually, in order to reduce the effect of outliers, the average of the last five speaker codes produced was used. This five frame smoothing window was also used on models produced using less input speech whose use will be described in subsequent paragraphs.

18. Using this wider window seemed justified in this case since a great deal more training data — from all one hundred and eighty nine training speakers — was being used.

19. Except, of course, that from the sentence sa_1.

speech. As before, distances were calculated by aligning every "training set" synthetic utterance with every *actual* "training set" utterance. Since there are one hundred and ninety speakers in the training set, this involved more that thirty six thousand alignments.

For training set speakers, on average, the mean frame distance on the best alignment path between the synthetic speech and the real speech it was aimed at mimicking was 0.38 lower than the distance from speech from other speakers. For testing speakers, the effectiveness of the transformation was lower; the distance was 0.26 less, on average, for the target speaker than for other speakers.

The results of the test with a simpler speaker model and fewer speakers were confirmed. The speaker model was allowing the transformation network to move the synthetic speech in the direction of the target speaker's voice.

6.9.2. The time course of transformation quality

To get an idea of how rapidly the useful information became available in the speaker code, the same procedure was repeated, using speaker codes extracted after fewer phones had been heard. The results are summarised in Table 45. For training speakers, the effect of the

Table 45: The influence of amount of speech used to form the speaker code on the quality of the voice transformation achieved. The measure given is explained earlier in the chapter. Larger values suggest better conversion.

Number of phones used to form speaker model	5	15	50	100	200
Train	0.332	0.371	0.380	0.373	0.379
Test	0.228	0.242	0.283	0.270	0.265

speaker model was fairly stable for all models built with more than five phones. For the testing speakers, the fall off in quality was perhaps more gradual, with an apparent decline for the models formed from fifteen phones or fewer.

Although it is fairly clear that five phones were not enough input to allow the speaker model to reach the final speaker code, there was enough variation among codes formed with more speech to prevent any trend beyond that from being evident.

6.10. Validating the final system with human listeners.

Results presented above showed that transformations controlled by a speaker model could reduce the distance between the output of the synthesiser and the speech of the target speaker, when that distance is measured by spectral distortion.

It was also important to know whether this measured distance reduction corresponded to a reduction in perceptual distance. Did the transformed speech from Software DECtalk sound more similar to the voice of the target speaker than the original dectalk speech did? If so, did it sound more like the voice of the particular speaker it is intended to imitate than the voice

of some randomly chosen alternative speaker? The following experiment was intended to provide initial answers to these questions. The aim was to determine whether listeners can identify which of a pair of target speakers, an utterance generated on the output of the transformation is intended to mimic the utterances of a particular target speaker. This experimental design loosely follows the practice of the papers from Abe *et al.* described in this chapter's introduction.

6.11. Experiment: Human ability to discriminate transformed voices.

In this experiment, unaltered Software DECtalk speech was viewed as being at one end of a continuum and the speech from the target speaker was viewed as being at the other end, with stages in the transformation of the former into the latter lying between them. The aim was to measure the degree to which each step in the conversion process affected the perceived voice personality of the speech. These steps are illustrated in Figure 41. Measure-

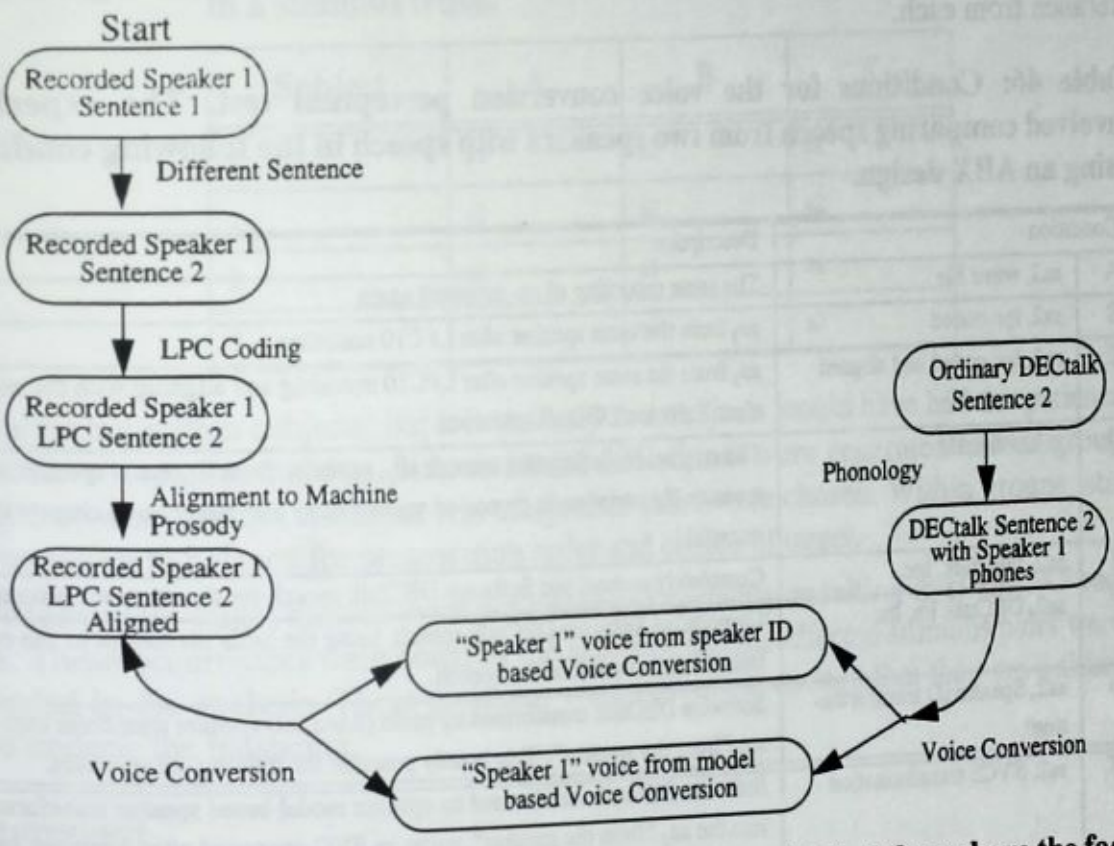

Figure 41: Transformation stages for voice conversion. At the left, we have the factors that increase the distance between the actual target utterance and the output of the voice conversion network. At the right, the factors that increase the distance between the input we would like to have to the voice conversion, and that we do have

ments were made of the ability of listeners to identify which of two speakers, introduced with samples raw speech — labelled "start" in the diagramme — was the target speaker of the transformation. Measurements were made of subjects' ability to identify the target speaker from the speech output from every step in the conversion, in an effort to get an indication where the greatest changes in speaker personality occurred.

6.11.1. Method

Materials

Using the transformation steps shown in Figure 41 as a guide, utterances for a variety of processing conditions, listed in Table 46, were produced for four sets of thirty six target speakers. These sets of thirty-six were composed of nine male and nine female speakers, chosen at random from each of the training and testing speaker sets. Speakers could be reused between sets of thirty six speakers, but not within. Within sets, the training speakers were divided into three groups of three pairs of speakers: three pairs of males, three pairs of females, and three pairs of whom one was male and one was female. Test speakers were similarly divided.

The aim of the experiment was to determine the rates at which speakers in each of the six sets of three pairs could be distinguished from each other, for each of the listed processing conditions, to measure the degree to which voice personality was retained by the conversion process, and, as a base line, the rates at which speakers could be distinguished given a short utterance from each.

Table 46: Conditions for the voice conversion perceptual test. The experiment involved comparing speech from two speakers with speech in the following conditions, using an ABX design.

Condition		Description
A	sa2, wave file	The same recording of sa_2 repeated again.
B	sa2, lpc coded	sa_2 from the same speaker after LPC10 encoding
C	sa2, lpc coded and aligned	sa_2 from the same speaker after LPC10 encoding and aligning with the equivalent Software DECtalk utterance.
D	sa1, wave file	The original recording of a second, sa_1, sentence from the same speaker, to measure the variation in perceived voice quality due merely to a change in material.
E	sa2, DECtalk, lpc	Completely unmodified Software DECtalk speech for the same sa_2 utterance
F	sa2, DECtalk Ph, lpc	Unmodified Software DECtalk speech, using the same phonemes as the original speaker to produce the speech.
G	sa2, SpeakerID transformation[a]	Software DECtalk transformed by multi (fixed set) speaker transform into the sa_2 "from the speaker". This is only possible for training set speakers.
H_1	sa2, SVC5 transformation	Software DECtalk transformed by speaker model based speaker transform into the sa_2 "from the speaker", using an SVC generated after 5 phones had been heard, and smoothed over a 5 phoneme window starting from that point.
H_2	sa2. SVC15 transformation	*ibid*, but after 15 phones heard
H_3	sa2. SVC50 transformation	*ibid*, but after 50phones heard
H_4	sa2. SVC100 transformation	*ibid*, but after 100phones heard
H_5	sa2. SVC200 transformation	*ibid*, but after 200 phones heard
H_n	Novel, SVC200 transformation	and, a completely novel sentence outside the set used in the TIMIT database, transformed using the SVC produced after 200 phones had been heard.

a. Since speaker ID inputs were not, of course, trained for the testing group speakers, this utterances in this condition were only produced for the eighteen training speakers.

Within each of the four speaker sets, two hundred and twenty five ABX stimulus triples were generated, using techniques covered in detail earlier, from the eighteen speaker pairs. In each case the A and B stimulus were down-sampled[20] versions of the original (16 bit, 16kHz) recording of sa_2, *"Don't ask me to carry an oily rag like that"*, from two speakers A and B, and the third sample, X, was generated under one of the conditions of Table 46. Which of the two speakers in a pair were assigned to be A or B, and which of A or B would be used to generate the "matching" stimulus X was chosen randomly for each stimulus. Even the unmodified Software DECtalk speech had one of the target speakers randomly assigned as a "correct" match. To minimise the effect of any bias from the particular choice of X to be deemed correct, and of a listener bias in favour of the second utterance, stimuli were generated for subjects in groups of four, with the only difference being that the presentation order within each triple and which of the reference speakers "A" and "B" would be used to generate the test stimulus, as shown in Table 47. It would be better to do this counterbalancing within subjects, but it is unlikely that subjects would have had the patience for a six hour experimental run. Sixteen sets of 225 stimuli were generated in four groups of four. Between groups, different sets of speaker pairs were chosen. Within groups, stimuli were chosen to balance for presentation order and choice of target.

Table 47: Permuting the triples across subjects. Entries are target speakers used or each of the four subjects in a quad, for each position in a stimulus triple.

Subject	A	B	X
1	s1	s2	s1
2	s1	s2	s2
3	s2	s1	s1
4	s2	s1	s2

Occasionally target stimuli were not generated correctly, resulting in silent "X" recordings. These occurrences were noted by the subjects, and the affected stimulus pairs were not included in the analysis. These problems were infrequent enough that they are unlikely to have undone the balancing.

Equipment

Stimuli were presented monaurally through the left ear using identical headsets plugged into the headset outlets of Digital Equipment Corporation Alpha work stations. Playback was managed using the freely available "Audiofile" audio presentation system [ref]. Raw recordings were down-sampled to 8kHz immediately before playback, and LPC-LAR recordings were decoded immediately before playback by the modified LPC-10 coder used throughout the thesis. Stimuli were presented to subjects, and subjects' judgements

20. To 8kHz so that it could be presented using "Audiofile".

recorded, using the user interface shown in Figure 42. This user interface was written in the

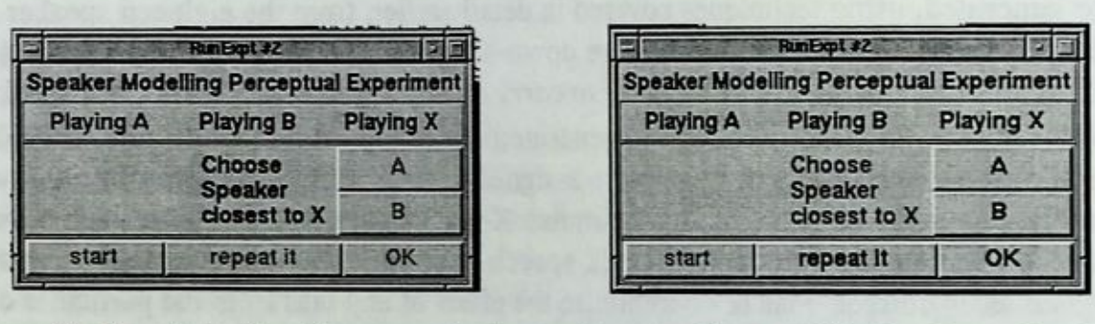

Figure 42: User interface for perceptual experiments, shown while the first stimulus in a triple is playing, and after the speaker has made an initial match X=A.

Tk/Tcl language.

Subjects

Sixteen volunteer subjects in their mid-twenties to early thirties were used. Most were CMU computer science graduate students. Fourteen subjects were male, and two female. No special attempts were made to keep the purpose of the experiment from the subjects.

Procedure

Subjects were seated before one of three alpha work stations displaying the user interface in Figure 42. They were read the introductory passage given in Appendix E., and then asked to start the experiment.

For each stimulus triple, the subject would hit the "Start" button. The three stimuli were played in order, with the user interface indicating which stimulus, A, B, or X was being played by highlighting the appropriate label in green, as shown on the left side of Figure 42. After all three stimuli had played, it became possible to make a choice between A=X or B=X, as shown in the right hand side of the figure. This choice could be changed, or the stimulus triple represented, repeatedly until the "OK" button was presented. None the subjects had any difficulty completing the task.

6.11.2. Results

Table 48: Results of the first perceptual experiment. Each cell contains the percentage of correct identifications of stimulus X as matching the voice in stimulus A or B. These numbers are for three target speakers in each condition for sixteen subjects, giving 48 trials per cell[a]. Means in the bottom row are calculated over the SVC model based conversions indicated by background shading.

Condition		Train			Test		
		Both	Female	Male	Both	Female	Male
A	sa2, wave file	100.0	100.0	97.9	97.9	97.9	97.9
B	sa2, lpc coded	100.0	95.8	89.6	97.9	95.8	93.8
C	sa2, lpc coded,aligned	97.9	89.6	77.1	100.0	85.4	81.3
D	sa1, wave file	97.9	87.5	83.3	97.9	83.3	89.6
E	sa2, DECtalk, lpc	52.1	58.3	43.8	52.1	47.9	37.5
F	sa2, DECtalk Ph, lpc	45.8	60.4	47.8 (46)	60.9 (46)	38.6 (44)	56.3
G	sa2, SpeakerID transformation[b]	91.7	56.3	60.4	-	-	-
H_1	sa2, SVC5 transformation	66.7	47.9	43.8	81.3	41.7	56.3
H_2	sa2. SVC15 transformation	81.3	50.0	52.1	66.7	39.6	58.3
H_3	sa2. SVC50 transformation	79.2	52.1	56.3	77.1	43.8	60.4
H_4	sa2. SVC100 transformation	79.2	47.9	50.0 (46)	81.3	56.3	52.1
H_5	sa2. SVC200 transformation	76.1 (46)	48.9 (47)	64.6	68.8	56.3	56.3
H_n	Novel, SVC200 transformation	77.1	48.9 (47)	54.2	81.3	39.6	52.1
	Mean	76.5	49.4	53.4	75.0	47.5	56.7

a. As noted above, a few of the X targets were not properly produced, reducing the cells marked to the number of trials shown in brackets.

b. Since speaker ID inputs were not, of course, trained for the testing group speakers, this utterances in this condition were only produced for the eighteen training speakers.

Subject's selections of "A" or "B" as the match for "X" were compared with the correct assignment recorded during stimulus preparation, and the proportion of correct selections tabulated in Table 48. It is clear that the task is not an easy one; Even on the task of telling which of the original two utterances had been repeated exactly (row A in the table), subjects did not perform perfectly, and by the time the speech of the target speaker had subjected to lpc encoding and decoding (B) and then aligned with the Software DECtalk speech (C), they were having considerable difficulty telling speakers apart, choosing correctly in about 85% of the cases for pairs of female, and about 80% of the time for pairs of male speakers. The basic encoding used for speech in this transformation was having a substantial effect on the perception of voice personality differences, even when the voice transformation had not been applied.

The difficulty of making voice personality distinctions at all, given a single phrase from each speaker, is illustrated by condition D, where the speakers had to choose which of the two utterances of *"Don't ask me to carry an oily rag like that."* matched the voice that had

uttered *"She had your dark suit in greasy wash water all year."* Subjects made a great many errors telling speakers apart in pairs of matched gender, even though they were listening to clean speech.

It is worth noting that in almost all these conditions with untransformed (but, in some cases, altered) speech, subjects found it easier to distinguish pairs of male test speakers than to distinguish pairs of male training speakers, and that they found it easier to distinguish women than men. While one could easily imagine that there might be gender based differences in the degree of voice personality, the source of the difference between the training and test set remains somewhat mysterious.

Conditions E and F were included as controls. In condition E, raw lpc processed Software DECtalk speech, was presented in the X condition, using phonemes chosen to match those of the target speaker. In condition F, the utterance X was the completely unaltered lpc encoded Software DECtalk speech, regardless of who A and B were. One would expect performance in condition E to be almost completely random, and on F completely random. The fact that some cell values skate dangerously close to significance reminds one of the dangers of reading too much into any individual cell of a large table. Pooled across all conditions in E and F, however, there was no evidence that listeners could guess above chance (P(rate of correct guessing > 0.5, 568 trials) = 0.48).

Conditions $H_{1\text{-}7}$ and H_n were, of course, the focus of the experiment. To deal with the obvious observation first, the speaker model clearly imposed enough voice personality on the transformation to enable men and women to be told apart in many cases — correct identification rates for all the speaker model conditions for pairs of mixed gender were significant at or above the 2% level. What is perhaps more surprising, given the clear separation for gender of the speaker models when plotted in a previous chapter, is that gender could not be separated more reliably. Within pairs of the same gender, the effect of the model is subtle. There was clearly no information retained by the transformation that allowed subjects to tell women apart. For men, though, there was some evidence that the model-based voice transformation was imposing some personality. Although the evidence is not overwhelming for models generated from any of the particular amounts of speech (5, 15, 50, 100 or 200 phones), when results were pooled over all model transforms, within the male speaker pairs, the probability that subjects performing were performing at or below chance was less than 10% for the training speakers, and less than 3% for the test speakers. And, as noted above, the test males seemed to be intrinsically easier to distinguish.

For the sake of completeness note that, as one would expect, since they were played nearer in time, there was a slight but highly significant bias in favour of matching the second (B) utterance of ABX triple with X, (46.3%A 53.7%B). This bias was completely controlled for across speakers by the balanced design of experiment, except for a possible slight effect from the trials with missing X.

It is also worth noting, for the sake of informing the course of future experimentation, that many of the subjects in the experiment mentioned that they were using mainly prosodic cues to match utterances — cues which the current transformation could not possibly capture.

6.11.3. Discussion

It is clear that the speaker model tested included sufficient information about the speaker to specify their gender in many cases. Beyond this the evidence is less clear. It is possible that other salient speaker differences were represented form men, but not for women. Part of the difficulty in determining this is due to the fact that the voice transformation developed here was not as sensitive instrument for measuring the effects of speaker codes as one might have hoped. A discussion of possible reasons for this will be deferred to the end of the chapter.

Since there was clear evidence only that the speaker code allowed gender identification, and weaker evidence that it differentiated usefully between some male speakers, it seemed possible that it was simply affecting pitch. Now, it is true that it is important for such a model to capture pitch, as Valbret *et al* point out [valbret92b]:

> *"The average level of the fundamental frequency is a crucial factor [in voice personality]. Even on nonsense words, the average pitch-value seems to be the most important factor for speaker identification: spectral transformation without the correct pitch modification results in a voice that is not recognised as the target voice; on the other hand, pitch modification without any spectral transformation significantly improves the speaker recognition rate."*

However, one can measure average pitch by less involved means that the models investigated here. It was useful to investigate whether the voice transformations modulated by the speaker code were doing anything beyond affecting the average pitch.

6.12. Experiment: Is the effect of the speaker code accounted for by pitch changes?

6.12.1. Method

To investigate this question, another set of trials closely resembling condition H_4 in the previous experiment was run. The experimental materials were slightly different this time. Utterances A and B were derived from samples of speech from two different speakers uttering the sa_1 sentence. This time, though, this speech was LPC encoded and time aligned with Software DECtalk speech for the same sentence. After alignment, everything but the pitch of utterances A and B was replaced with data from the Software DECtalk speech, so that *only* pitch differences between the speakers could be used to select between them. As in condition H_4 above, the utterance X in the ABX design was Software DECtalk speech for another utterance (sa_2), transformed using the voice transformation system, using the speaker code for speaker A or speaker B. The same speaker model as before was used to generate the speaker code after one hundred phones had been heard from the speaker. If the transformation was affecting only pitch, subjects in this experiment should be able to match the speaker for utterances in A and B with the target speaker for the speaker in X as well as they had in the previous experiment.

There were three experimental conditions: comparison of two men's voices, comparison of two women's voices, and comparison of a woman's voice with a man's. All conditions were presented in both possible orderings of the voices used for A and B, and both choices of whether X would match A or match B.

Stimuli were presented using the same user interface as before, and the experiment was introduced using the same preamble with only the number of trials changed to eighteen. Subjects were twelve male members of the CMU Computer Science Department in their early twenties to thirties.

6.12.2. Results

Results for this experiment are given in Table 49. The first row of the table gives the rate at

Table 49: In this experiment, the effect of the ability of the speaker models to affect voice personality by altering pitch was investigated. The A and B stimuli were Software DECtalk speech with the pitch contour replaced with that from two human speakers, one of whom was the target of the voice transformation that produced stimulus X. Each cell contains the percentage of correct identifications of stimulus X as matching the voice in stimulus A or B. These numbers are for three target speakers for twelve subjects, giving 36 trials per cell.

	Train			Test		
	Both	Female	Male	Both	Female	Male
Percentage correct	77.8	47.2	63.9	69.4	50.0	41.7
Number of standard deviations from chance	3.33	-0.33	1.67	2.33	0.00	-1.00

which the subjects were able correctly to identify which of the speakers A and B the speech in X corresponded to. As in the previous experiment, the speaker models were clearly able to produce an effect on the synthetic voice that allowed subjects to identify the speaker's gender in many cases, with only the pitch of the reference speakers available as a cue to their identity. As one might expect, this effect was greater for the training set than for the test set. For the training set, there was also some evidence that the model was providing information that allowed male speakers to be distinguished on the basis of pitch, but, unlike the similar trend in the previous experiment, this effect did not generalise at all to the test set. It is possible that the model was learning to set parameters for particular male training speakers from the speaker model that made the speech match the targets better, but if it did so, it did not use the speaker models as positions in a speaker space into which test-set male speakers could usefully be placed. Where speaker identification was possible using these stimuli, it was performed at rates that were not dissimilar to those when spectral characteristics of the reference speakers were available for comparison with the output of the speaker model. If more information than pitch was being used in the previous experiment, it was only apparent in the case of male testing set speakers, and that evidence was very weak.

6.12.3. Discussion

As far as it was possible to tell, using the instrument provided by the voice transformation networks, the only perceptually salient information that was consistently encoded by the speaker models was information related to gender, and that information did not have a perceptible effect on anything but the pitch of the talker's voice.

6.13. Experiment: Speaker information apart from pitch?

6.13.1. Procedure

In a final attempt to see whether there were modelled voice qualities apart from pitch that could be used to distinguish speakers, an experiment was run using stimuli in which all differences in pitch had been removed. As before, stimuli were presented in an ABX setup, with each subject listening to nine pairs of women, nine pairs of men and nine pairs of men and women from each of the training and test set, for a total of fifty-four comparisons per subject. The order of presentation of the stimuli, and which corresponded to the X stimuli, were counterbalanced across the four listeners used.

The materials for the A and B stimuli were prepared by taking natural, LPC-coded, speech for sentence sa_1 and time aligning it to the same sentence spoken by Software DECtalk. After alignment, the pitch signal in the speech was entirely replaced by that from the Software DECtalk version, yielding samples all of which had identical pitch contours. The X stimuli was generated as in the previous two experiments, by producing the sa_2 sentence using the voice transformation network with the speaker code the selected speakers out of those who produced sample A and B. After this transformed utterance had been produced, its pitch contour was also replaced with that of the input Software DECtalk speech.

Materials were presented to subjects using the previously described interface, and the subjects were read the usual preamble, with the number of ABX triples replaced by "fifty-four".

6.13.2. Results

Results for this experiment are given in Table 50. The first row of the table gives the rate at

Table 50: In this experiment, the ability of the speaker models to affect voice characteristics other than pitch was investigated. Pitch contours for all three stimuli were replaced with that from Software DECtalk, forcing subjects to use other cues if possible. Each cell contains the percentage of correct identifications of stimulus X as matching the voice in stimulus A or B. These numbers are for nine target speakers for four subjects, giving 36 trials per cell.

		Train			Test	
	Both	Female	Male	Both	Female	Male
Percentage correct	50.0	41.7	63.9	58.3	44.4	44.4
Number of standard deviations from chance	0.00	-1.00	1.67	1.00	-0.67	-0.67

which the subjects were able correctly to identify which of the speakers A and B the speech in X corresponded to. With pitch differences removed, subjects reported that it was very difficult to tell speakers A and B apart, let alone to tell which of them corresponded to the target of X, and this is reflected in the Table. In none of the conditions were listeners able to tell which of speaker A and B was the target in X at rates significantly greater than chance, although in the case of male training set speakers they came close. If any information beyond pitch is contained in the speaker codes, it is either lost during the transformation, or the speakers are unable to use it once the pitch and timing components of voice personality has been destroyed.

There was some weak evidence that some voice personality beyond pitch was retained for male training set speakers, but if it is, future work on improved speaker models, and particularly on improved voice transformation networks will be required to demonstrate the fact conclusively.

6.14. General Conclusions from the Voice Transformation work.

Although the experiments in which spectral distortion for transformed speech were measures indicated that the speaker codes were allowing the transformation to move the Software DECtalk speech towards the target voice, human beings were not able to detect the effects of this movement except in as much as the pitch of the voice was concerned. Although pitch is surely an important component of voice personality it is important to extend the voice codes to include prosodic qualities such are relative segment duration, and to ensure that they accurately represent long term spectral characteristics beyond pitch. It is also important to improve the transformation so that it produces high quality synthetic speech, and so that it accurately expresses the information contained in the speaker code. Since there was considerable discussion of the quality of the speaker codes in earlier chapter, the discussion here concentrates on the voice transformation.

In [abe91a] the authors make an interesting observation that the codebook size of reproducing speech from two speakers accurately must be approximately twice as large as that for as single speaker. While this rate of increase may level off with more speakers, it suggests that the multi-speaker voice transformation problem is, in fact, much more difficult than the same problem with single speakers. Certainly that possibility is supported by the quality of the speech output by the voice transformation system used here, which presented a considerable barrier to its use in evaluating the speaker models used with it in perceptual experiments.

It is difficult to know how far the plurispeaker transformation has to be improved before it matches the quality of other systems in the literature. Savic and Nam said of their voice transformation system that *"Experimental results [not included in their paper] demonstrated that there was almost no difference between the target voice generated by the voice transformation system and the target voice output from the LPC Vector Quantisation Vocoder, which was used as a reference."* On the other hand, In the case of the system in the literature [abe90, abe91a] that had goals most similar to those of the current work, although the training data used was more friendly to alignment, the authors seemed to have reservations about quality:

> *"In terms of the converted voice quality, cross-language voice conversion is not as effective as voice conversion between Japanese speakers. One reason for this may well be that in the cross-language voice conversion experiment MITalk speech was used instead of human speech"* [abe91a].

It would have been very useful, when first building the system, to have used the same kind of neural network and speech representation to build a transformation between two human speakers and to do so using very short utterances as has been done in the literature. Since both the speaker models and the test applications were being developed together, there simply wasn't time to gather the data for this experiment and to train such a transformation. It should however be done. If the transformation in such a system proved to be of low quality, in contrast to those reported in the literature, the cause would plainly be due to the speech representation, or to the use of a neural network functional approximator instead of a codebook based-mapping or connectionist classifier. If the voice quality was high, the system would provide a gold standard from which one could proceed to replacing the source speaker with soft-talk, and thence to the synthesis of multiple target speakers.

Earlier it was pointed out that one of the greatest problems in building the transformation lay in generating a good alignment between the source and target speech. When the alignment is imperfect, the network it trained to transform an input frame into the mean of the target frames to which it is aligned, some of which will be completely inappropriate. The end result of this being "blurred" frames being fed into the resynthesis system, and the production of distorted speech. Obtaining a good alignment was made difficult both by the use of synthetic speech as one of the signals to be aligned, and by the fact that the system depended on aligning entire sentences. It may be possible to improve this alignment by borrowing an idea from the recirculating speaker models. If the transformation moves the source speech towards the target speech, the transformed source speech should be easier to align with the target utterance. By training the system, using this method to obtain a better

alignment, and retraining, iteratively, a sharper transformation should be able to be trained. Another possibility for improving the transformation may be to use a great deal more training data per speaker, obtained from a database other than TIMIT, and to simply discard sections for which poor alignments are obtained from the training data.

A further difficulty with the voice transformation, which is shared by those reviewed in the introduction, is that there are components of voice personality that it can't currently model. Instead of simply transforming the data on a frame by frame basis, future systems should cover all the components of voice personality, beginning with the choice of the correct phonetic realisation for the target speaker of the lexemes in the utterance, and ending with the adjustment of relative durations of the phones within those lexemes, or even of the pitch and loudness profile of the utterance as a whole. Although this is certainly an ambitious goal, it is also a necessary one. In the perceptual experiments here, which compared whole utterances, subjects often commented that they had used prosodic, rather than spectral, qualities of the utterances to match speakers.

Although the quality of the target voice representations and of the transformation used to express them was far from ideal in this initial implementation, it seems likely that both can be improved with further science, to make more of the variation explicit, and further engineering to express it. More will be said on those matters in the next chapter; this one is closed with sentiment the author would like to heartily endorse, with respect to the current work in multi-speaker synthesis with conversion:

> *"Because cross-language voice conversion is a very new idea, and also a very difficult problem, we would like to claim that we have at least shown the possibility of such conversion and demonstrated a possible method"* [abe91a]

Chapter 7. Conclusions and Future Work

Although none of the systems investigated in this thesis was a complete success, a good deal was learned about the speaker modelling enterprise itself, and about the prospects for applying such models in real world tasks. Perhaps the most important lesson was that doing work in this area is currently very challenging. It was necessary to build both the speaker models, and the mechanisms for testing them, and neither of these tasks were straightforward. If there had been a pre-existing speech recognition system or voice transformation system that was known to show clear performance improvements when told which of a large set of speakers it was dealing with, then a larger number of possible speaker models designed to distil that identity into a point in speaker space could have been developed and evaluated, increasing the likelihood of success. If there had been a body of work in developing free-standing models of speaker variation, then there would have been both established criteria for evaluating the current models, and a those models could have been applied to the chosen applications to provide a clear baseline of performance. Instead, the models and the applications had to be pull each other up by their bootstraps, a clumsy and imperfectly executed manoeuvre.

Despite the inadequacies of the models developed here, and despite the difficulty using them with systems that do useful work, the *idea* of developing speaker spaces and using them to help speech systems adjust to new voices seems as promising as it did at the beginning of this work.

There is a great deal of work to be done if this idea is to be realised. The following paragraphs will summarise the conclusions that can be drawn by the work reported here in the areas of speaker modelling generally, and the application areas in speech recognition and synthesis. They will also outline plans for future work that may be useful in approaching the goal of producing systems that can use their knowledge of the way voices vary to improve their performance in the face of the great variety of human voices.

7.1. Speaker models

The speaker models that were built satisfied many of the stated design requirements. They were compact, text independent and formed rapidly. They also captured important characteristics of the speakers, as demonstrated by the fact that speaker gender was visible in the code, and by the fact that they could reduce the distortion between the output of the voice transformation system and speech from the speaker represented by a speaker code.

The failure of the models to be useful in speech recognition was forgivable, since the full-scale recognisers were unable to make use of speaker identity at all. What was more disappointing was the lack of clear evidence from the voice transformation work that the speaker models had captured perceptually relevant variation that could not be accounted for by the pitch of a speaker's voice. Nevertheless, the fact that reasonably good speaker classification accuracies could be attained using the SVCs for nearest centroid classification and the high correlation between speaker models produced from different amounts of speech from the same speaker both support the suspicion that more information about was present in the

speaker models than the transformation application was capable of revealing. Ways of improving on the transformation to produce a more sensitive instrument for making explicit the content of speaker codes will be discussed below, but even if one accepts that the speaker models developed here have made a decent start at a representation of speaker variation, there are clear steps that should be taken to improve them.

7.1.1. Improving segmental models

In the Chapter 3, when the segmental models that combine to make the speaker models were discussed, there was some discussion of methods for normalising the duration of states within speech segments, so that a component of the input to the segmental models would correspond to a spectral channel and a state, rather than a time. Two methods of doing this were discussed: DTW alignment to a set of reference templates, and using states identified by a Markov model based speech recogniser. In the work here, no such normalisation was done — segments were reduced to identical size by linear time warping. While differences in the relative duration of states within phones may well be important to voice personality, it would be probably be better to model this explicitly, by including a vector of relative state duration, along with the set of state spectra, in the input to the segmental models. A comparison should be made between phoneme models produced this way with the current set should be made, to see to what extent explicit modelling of timing variation reduces the intra-speaker stability of the phoneme models.

The other component of the variation in phone models that should be made explicit is the variation due to phonetic context. This context has a strong effect on the way a phoneme is realised, but has nothing to do with speaker variation. Ideally, one would control for this by modelling speaker variation in every context-dependent phone separately, but lack of data and the difficulty of combing the results into and over all speaker model preclude this path. However, preliminary experiments suggested that neural networks could be used to estimate the effect of this variation on a phone within an additive model. This estimate could then be used to control for the context effect when measuring the differences between a phone uttered by different speakers. Phone models that include such a control for context effects ought, again, to improve the stability of phone models within speakers, and would be well worth constructing.

7.1.2. Improving overall speaker models

In general, the linear, statistical speaker models performed as well at forming speaker codes that distinguished speakers as the neural networks did, and the discriminative models, as one might expect, formed more distinctive codes that the "variational" or compression models The sole exception was the recirculating neural network model, which despite being a compression network, produced codes that distinguished speakers well.

If the phoneme codes provide more information about voice characteristics together than they do separately — if they are more than just linear combinations of each other — the neural networks ought to have been able to produce more compact speaker codes than the linear methods. It was certainly demonstrated, on toy problems, that the networks are capable, under ideal conditions, of producing much-better-than-linear encodings.

One reason for this promise not being realised might simply have lain in the fact that the phone models were very noisy and this noise may have masked the interphone correlations the networks needed to observe. If this was the case, simply improving the phoneme models might be sufficient to give a modelling advantage to the neural networks. In any case, to ensure that the neural networks do at least as well as the linear methods, and that training is not expended in learning an imperfect linear model, they should be pre-loaded with weights derived from a linear model before training begins.

Beyond whatever improvements can be made to the raw modelling technology, there is still the matter of perceptual relevance. It was not possible to demonstrate conclusively that perceptually relevant voice properties beyond speaker gender were retained by the current model. It should be possible to improve on this situation. If a very large number of judgements of the degree of similarity of pairs of human voices are gathered from human listeners, the technique of multidimensional scaling can be applied to place these voices within a space in which distances between voices correspond to human perceptual distance. Although the effort involved in collecting the large number of similarity judgements needed would be considerable, the speaker model produced would be valuable; it would provide a standard against which other speaker models could be compared, and the codes representing the position of a speaker in this space could be used as training targets for models, like the present ones, derived automatically from the speech itself.

7.2. Speaker models for speech recognition

Although the application of speaker models to speech recognition here was unsuccessful in almost every respect, this failure cannot be ascribed to the speaker models. In the recognisers that were able to use speaker information at all, namely the recognisers for the Peterson and Barney data, task independent speaker models provided about the same amount of information about sex and age. Of course, one would hope for more than that from a general model of speaker variation. The measures described in the last section intended to decrease the amount of irrelevant variation are likely to improve the quality of these general models, but even of that effort remains relatively unsuccessful, there remains reason for hope. When voice information about a speaker was made available to the Peterson and Barney recogniser, by making the formant values from other phonemes spoken by the same speaker available through a bottleneck, the classifier was able to use this information to greatly increase its recognition performance. In light of the results presented in this thesis, the most likely path to success in applying speaker models to recognition lies in building models that are general, in as much as that they work for new speakers, and do not require retraining, but which are trained in the context of the particular recogniser in which they will be used.

That said, the major problem with applying speaker modelling to speech recognition was that an attempt was made extend to use speaker information in connectionist classifiers to realistic recognition tasks met with very little success. Even recognisers given perfect information about speaker identity benefited little from that information, precluding the possibility of large gains from imperfect information derived from speech. Given that speaker specific recognisers still outperform speaker independent ones, and that adaptation schemes involving additional training of parts of a recogniser with a sample of speech from a new speaker are generally somewhat successful, it is important to explore why speaker ID was

not beneficial. A start was made here: part of the difficulty lies in ensuring that one is providing the recogniser information that it cannot obtain elsewhere. The connectionist recognisers used here, despite their imperfect classification performance, had the advantage of being able to use a wide input window. The experiments with reducing the amount of information visible through this window suggested that the much of the information that could have been derived from speaker ID was already available in the multi-frame input. It also appeared that part of the difficulty was due to the homogeneous nature of connectionist classifiers, in which the use of speaker information to improve vowel classification appeared to interfere with the same classifier's ability to correctly recognise consonants.

If rapid adaptation using speaker models, whether those models are task independent or recogniser specific, is to fulfil its promise, then it will be necessary to gain a far better understanding than we presently have of what recognisers actually do with speech with different speakers, and how this causes errors to be produced. Then it will be clearer what prior information about a speaker's voice could be used to prevent the errors, and when. It may be more productive to pursue such an investigation using a Markov model recogniser, such as Sphinx, where the model parameters have more transparent roles than the weights in a neural net. In Markov models, the distributions associated with inputs to be associated with a particular acoustic state are explicit. If there are many misrecognitions associated with particular examples of these states in speaker recognition mode, then one can compare the distributions for the individual speaker and see whether, for example moving the means of the reference distributions for the state would suffice. If so, and if these mean shifts were correlated across acoustic states for a speaker, one would have a good idea of what sort of speaker model — in this case a regression model between deviations from acoustic state means — is likely to be productive.

7.3. Speaker models for voice transformation

. Using a single connectionist network as a functional approximator, it was possible to transform frames of input speech into something more closely approximating the voice of a target speaker, reducing the pitch and spectral distortion between the synthetic source speech corresponding speech from the target speakers. This was the case both where speaker ID was used to select the target speaker, and when speaker codes derived from one of the speaker models was used. When listening tests were done, however, the only information about speaker identity that was imposed with any reliability on the target speech was speaker gender, and that imperfectly. Despite the fact that the speaker information was affecting other components of the speech signal that pitch, it appeared that audible changes in fundamental frequency of the transformed speech were sufficient to account for the effects of the models in the listening tests. Given that pitch is such an important component of voice personality, this may not be entirely surprising. When samples of natural speech had identical timing[1] and pitch information imposed on them, there was very little perceptible difference between the voices of different speakers, especially speakers of the same sex. If the spectral changes produced by the transformation were related to voice personality,

1. Or as nearly identical as possible, given the difficulty of doing whole-sentence forced alignments, guarantees about timing are difficult to give.

they may have been rendered undetectable by the generally poor quality of the synthetic speech.

Disappointingly, in the light of the claims made for the similar voice transformation systems reviewed from the literature, the synthetic speech produced was, at best, barely intelligible.

Given that the voice transformation did serve to make the effects of speaker codes on the speech produced observable, work to improve the basic transformation system is likely to be rewarding. The main difference between the system used here and those in the literature was the relatively small amount of speech available for a particular target speaker, and the use of synthetic speech as input. It was clear that the alignments produced between the source and target training set were imperfect, even after the work that was done to improve them, and that misalignments in training data are likely to decrease the quality of the transformation that can be learned. Although time consuming, the only obvious way to find out how much an improvement in alignment can improve the voice transformation is to do the alignment by hand, for a larger amount of speech from a single target speaker, matching the quality of training data used in the systems in the literature.

If high quality transformed speech can be produced from synthetic speech by improving the alignment in training, a series of experiments needs to be performed to determine exactly which components of the target speaker it is most important to produce to transmit voice personality. Speech in which only the pitch has been changed, or the relative segmental duration, or the phonetic realisation of the lexemes, or the LPC coefficients, or particular combinations of all of them, should be compared for its ability to transmit voice personality. Only then will it be clear what sort of voice models need to be built to support plurispeaker synthesis well. It may turn out, for example, that relative segmental duration, a feature that was not included at all in the speaker models developed here, is one of the main features of perceived voice personality.

7.4. General Conclusions

Although technological artifacts, in the form of a recogniser with an improved ability to handle speech from a variety of speakers, or a synthesis system producing clear speech in a variety of voices, were not produced in the course of this thesis, the work done here should contribute to their production in the future.

The idea of quantifying the dimensions along which speakers vary is an important one. The models built here captured the variation in some of those dimensions, and pointed to others sources of variation that contaminate the current models, and which themselves need to be modelled.

The work in applications for the models showed that there is a great deal to be learned about the applications themselves. It is not clear, for instance, why recognisers with speaker identity information provided do not perform as well as ones that are trained on specific speakers — what are the parameters set in the latter case that cannot be modulated in the former? What are the components of voice personality that a transformation system *should* learn to affect and that a speaker model should describe? Building explicit models of inter-

speaker variation that address these questions will improve our understanding both of the problems of these speech technologies and of speech itself.

It is hoped that both by improving the task independent models so that they capture more of what is truly distinctive about a speaker's voice, and are less contaminated by what is said and how, and by working in specific domains to discover exactly what *is* distinctive about speaker's voices, the difficult and often frustrating start made here can be turned into a first step leading to the sort of universal models of speaker variation that were hoped for when this work was begun.

References

[abe88] Abe, M., Nakamura, S., Shikano, K. and Kuwabara, H. Voice Conversion through Vector Quantization. *ICASSP 88, Proceedings of the 1985 IEEE International Conference on Acoustics, Speech, and Signal Processing*. pp 655-8.

[abe90] Abe, M., Shikano, K. and Kuwabara, H. Cross Language Voice Conversion., ICASSP 90, *Proceedings of the 1990 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol 1. pp 345-8.

[abe91a] Abe, M., Shikano, K., Statistical analysis of bilingual speaker's speech for cross-language voice conversion. 1990, *Journal of the Acoustical Society of America* Vol 90 No 1, July 1991, pp 76-82.

[abe91b] Abe, M. A Segment-based Approach to Voice Conversion, ICASSP 91, *Proceedings of the 1991 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol 2. pp 765-8.

[abe93] Abe, M., Statistical Analysis of the Acoustic and Prosodic Characteristics of Different Speaking Styles. In *Procedings of the 1993 European Conference on Speech Communication and Technology*, pp 2107-2110, 1993

[artières93] Artières, T., and Gallinari P., Neural Models for Extracting Speaker Characteristics in Speech Modelization Systems", In *Procedings of the 1993 European Conference on Speech Communication and Technology*, pp 2263-2266.

[asoh90] Asoh, H,. and Otsu, N., An Approximation of Nonlinear Discriminant Analysis by Multilayer Neural Networks , 1990, *IJCNN International Joint Conference on Neural Networks*, 1990. Vol 3, pp 211-216.

[assman82] Assmann, P. F., Nearey, T. M., and Hogan, J.T. Vowel Identification: Orthographics, perceptual, and acoustic aspects. *J. Acoust. Soc. Am.* **71(4)** April 1982, pp 975-989.

[ayer93] Ayer, C.M., Hunt, M.J. and Brookes, D.M., A Discriminatively Derived Linear Transform for Improved Speech Recognition. In *Procedings of the 1993 European Conference on Speech Communication and Technology*, pp 583-586, 1993

[becker88] Becker, R. A., Chambers, J. M. and Wilks, A. R. *The new S language*. Wadsworth & Brooks/Cole, Pacific Grove, California. 1988.

[bimbot92] Bimbot, F., Mathan, L., Lima, A., and Chollet, G. Standard and Target Driven AR-vector Models for Speech Analysis and Speaker Recognition. ICASSP 92, *Proceedings of the 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol 2. pp 5-8.

[blackburn93 Blackburn C., Vonwiller J. & King R. (1993) Automatic Accent Classification using Artificial Neural Networks In *Procedings of the 1993 European Conference on Speech Communication and Technology*, pp 1241-1244.

[blomberg89] Blomberg, Mats, Voice Source Adaptation of Synthetic Phoneme Spectra in Speech Recognition. In *Procedings of the 1989 European Conference on Speech Communication and Technology*,, pp621-624

[boulard88] Bourlard, H., and Kamp, Y. Auto-Association by Multilayer Perceptrons and Singular Value Decomposition. *Biological Cybernetics*, 59, pp 291-294, 1988.

[bridle91] Bridle, J.S. and Cox, S.J. RecNorm: Simultaneous Normalisation and Classification applied to Speech Recognition, in Lippmann, R.P., Moody, J.E. and Touretzky, D.S. Eds, *Neural Information Processing Systems 3*. 1991. Morgan Kauffman, San Mateo.

[chambers93] Chambers, J. M., and Hastie, T.J. *Statistical Models in S*, Chapman & Hall, London. 1993.

[childers85] Childers, D.G., Yegnanarayana, B. and Wu, Ke. 1985, Voice Conversion: Factors Responsible for Quality,. In ICASSP 85, *Proceedings of the 1985 IEEE International Conference on Acoustics, Speech, and Signal Processing*. Vol 2. pp 748-751.

[childers89] Childers, D.G., Wu, Ke., Hicks, D.M., and Yegnanarayana, B., Voice Conversion, 1989. *Speech Communication* 8 (1989) 147-158.

[conway88] Conway, J.H. and Sloane, N.J.A., *Sphere Packings, Lattices and Groups*, 1988 Springer Verlag, Berlin

[cottrell90] Cottrell, G. W., Extracting features from faces using compression networks: Face, identity, emotion and gender recognition using Holons. In Touretzky, D., Elman, J., Sejnowski, T. and Hinton, G.E. (Eds) *Proceedings of the 1990 Connectionist Models Summer School.* San Mateo: Morgan Kaufman. pp328-337.

[cox89] Cox, S.J., and Bridle, J.S. Unsupervised Speaker Adaptaion by Probabilistic Spectrum Fitting. In *Proceedings of the 1989 IEEE Interna tional Conference on Acoustics, Speech, and Signal Processing.* pp 294-297, April 1989.

[cox90] Cox, S.J., and Bridle, J.S. Simultaneous speaker normalisation and utterance labelling using Bayesian/neural net techniques. 1990. In *Proceedings of the 1990 IEEE Interna tional Conference on Acoustics, Speech, and Signal Processing.* Vol. 1. pp 161-4.

[cox93[Cox, Stephen. Speaker Adaptation Using a Predictive Model. In *Procedings of the 1993 European Conference on Speech Communication and Technology*, pp 2283-2286, 1993

[creelman57] Case of the Unknown Talker. *J. Acoust. Soc. Am*, **29**, 1957 p 655.

[dectalk94] Software DECtalk, Digital Equipment Corporation Maynard, Massachusetts, 1994.

[demers93] DeMers, D., and Cottrellm G. Non-Linear Dimensionality Reduction. In Hanson, S.J., Cowan, J.D. and Giles, C.L. (Eds) NIPS 1993. *Advances in Neural Information Processing Systems.* pp 580-587. San Mateo: Morgan Kaufman. 1993.

[dennis91] Dennis, S and Phillips, S. Analysis tools for neural networks. *Technical report (University of Queensland. Key Centre for Software Technology)* ; no. 207. May 1991.

[duda73] Duda, R.O. and Hart, P.E. *Pattern Recognition and Scene Analysis*, 1973, Wiley, New York.

[eskenazi93] Eskénazi, M. "Trends in Speaking Styles Research", *EUROSPEECH 93, Proceedings of the 3rd European Conference on Speech Communication and Technology.* Berlin. 1993. pp501-509.