AUTOMATIC SPEECH SYNTHESISER PARAMETER ESTIMATION USING HMMS

R.E. Donovan & P.C. Woodland
Cambridge University Engineering Department
Trumpington Street, Cambridge, CB2 1PZ, United Kingdom
red@eng.cam.ac.uk pcw@eng.cam.ac.uk

ABSTRACT

This paper presents a new approach to speech synthesis which uses a set of decision tree state clustered triphone HMMs to automatically segment a single speaker speech database into sub-word units suitable for use in a synthesiser. Parameters are then obtained for each of these sub-word units from the segmented database, enabling a basic synthesis system to be constructed. This automatic generation of synthesis parameters means that the system can easily be retrained on a new speaker, whose voice it then mimics. It also means that a very large number of sub-word units can be used, which enables more precise context modelling than was previously possible.

1. INTRODUCTION

This paper introduces a new approach to speech synthesis which uses triphone HMMs to segment a speech database into sub-word units. The system is completely automatic, and requires about 15 hours CPU time on an HP735-99 to convert one hour of single speaker speech into the parameters necessary to drive a synthesiser. The synthesised speech then mimics the voice of the speaker used to record the database.

There have been a number of previous attempts to use HMMs for speech synthesis, [1] - [3]. These differed from the current approach in that they used a single large HMM with 64 to 256 states to model all speech events. Synthesis from phonemes was only possible in [3], where another discrete HMM was used to calculate the most likely state sequence through the first HMM given the phoneme string.

The current system is made possible due to recent advances in speech recognition technology [4], [5]. It uses a set of cross word triphone HMMs, whose states are clustered using a set of automatically generated decision trees. These trees enable HMMs to be constructed for all possible triphones, many of which do not occur in the training data. Synthesis from triphones is therefore possible using the clustered states of the set of HMMs as the sub-word units. The automation of the system means that a very large number of sub-word units can be used in synthesis. With a 45 symbol phone set, approximately 90,000 triphones are possible. In current systems these triphones typically share 6,000 to 10,000 clustered states, meaning that this number of sub-word units are used in synthesis. This compares to about 2,000 sub-word units for a diphone system using the same phone set. This large number of sub-word units, and the context sensitive framework into which they fit enables more precise context modelling than was previously possible, and should lead to greater naturalness in the synthesised speech.

The automation of the system also means that the system can be easily retrained to produce any desired voice, provided training data can be obtained. The acquisition of specific new voices using traditional speech synthesis systems, if it is possible at all, is a very laborious process. In rule based systems, such as DECtalk, a large number of voice design parameters can be altered to change the voice. However, which settings to use to get a particular voice are far from obvious, [6], and producing exactly the voice required may not be possible. More recently diphone synthesis has been successful [7]. Here limited voice transformation can be achieved by scaling the formant frequencies, and altering the average pitch level. Obtaining a specific new voice is possible by preparing a new diphone dictionary using speech of the desired voice. However, this is a lengthy process involving manually splicing diphones from specially prepared carrier words and sentences [8].

2. HMM CONSTRUCTION

Initially a pronunciation dictionary and word level transcriptions are used to create a phone level transcription for each sentence in the training database. These transcriptions are then used to train a set of monophone models. These models are then cloned to produce a triphone model for every distinct triphone in the training data. The transition matrix remains tied across all the triphones of each base phone. The triphone models are then re-estimated.

For each set of triphones derived from the same base phone, corresponding states are clustered. This is necessary since there are usually too few occurrences of many of the triphones in the training data to properly estimate a Gaussian distribution for each of the model states. The clustering technique used also enables the construction of triphone models not present in the training data.

The decision trees that perform the clustering are constructed automatically using a large list of questions concerning immediate phonetic context, and two clustering parameters. A tree is built by asking every question at each terminal node in the tree, and splitting each node using the question which gives the largest increase in the log-likelihood of the data being generated from the tree, using a single Gaussian distribution to model the data at each node. The two clustering parameters specify the minimum number of frames of speech that must be assigned to each node, and the minimum increase in log-likelihood which must be achieved for a node to be split. The clustering procedure continues until these minima are reached. The final terminal nodes form the clustered states for that tree.

The decision trees are then used to calculate which clustered states to use to construct HMMs for all the possible triphones not present in the training data. Finally the state-clustered models are re-estimated. A more detailed description of the construction of decision tree state clustered HMMs is given in [5]

0-7803-2431-5/95 $4.00 © 1995 IEEE

The result of this procedure is a list of about 90,000 logical HMMs which typically share 6,000 to 10,000 clustered states, each of which is modelled using a single Gaussian distribution. Gaussian mixture distributions are not used in the current system since this would require more training data per state, and it is desirable to keep the amount of training data required to a minimum. A single Gaussian is considered to be adequate since the models are only used for alignment, not recognition, and the speech from only a single speaker is modelled.

Finally, the training data is aligned to the state-clustered models, producing label files of clustered state name against time. These label files are then used by the rest of the system to obtain the synthesis parameters.

3. OBTAINING SYNTHESIS PARAMETERS

All the occurrences of a particular clustered state in the label files produced above are pooled and the average duration and the standard deviation of the duration for that state are found. All the speech labelled as belonging to a particular clustered state is pooled to estimate the average energy per sample, the average zero crossing rate and the linear prediction (LP) coefficients for that state.

The average durations are, of course, based on fluent natural speech, which tends to be quite fast. Speech synthesised using only those durations is often too fast, and difficult to understand. Therefore each state is synthesised for an amount of time equal to the average duration plus a scaling factor multiplied by the standard deviation of the duration. This ensures that when the duration of an utterance is increased, states whose durations are most variable are stretched the most, and states whose durations are least variable are stretched the least.

When a state is being synthesised the average zero crossing rate for that state is thresholded to give a voiced/unvoiced decision to determine the excitation signal to be used, and each pitch pulse synthesised is scaled so that its average energy per sample matches the average energy per sample of that state.

Several methods for estimating the linear prediction coefficients have been investigated, and results from two of these, the P-method and the I-method, are presented in the next section.

The P-method pools all the speech labelled as belonging to a particular clustered state. A single autocorrelation vector is calculated from this pool, from which the LP coefficients are obtained. This can be shown to be the optimal way of finding the LP coefficients from more than one speech segment.

All other methods investigated involved selecting one particular piece of waveform to represent each clustered state. This is done principally to enable the possible future use of the PSOLA synthesis scheme, [9]. PSOLA works directly with the sub-word unit waveforms, and would be used in the system in place of the current LP scheme. This development could bring a substantial improvement to the synthesised voice quality in the future.

The F-method considers every frame that lies within the regions marked as belonging to a particular clustered state as possible candidates. The best frame is chosen by calculating the probability density of each frame's MFCC vector using the single Gaussian associated with each state. A 25ms Hamming window centred on the centre of the most probable frame is then used to calculate the LP coefficients. Occasionally frames are picked which are not good representations of a state, and this results in an audible glitch in the synthesised speech. The C-method is similar to the F-method, except that only the frames at the centre of each occurrence of each clustered state are considered as candidates. This method was tried in the hope that restricting the search in this way, ensuring that speech is not chosen from the boundary regions of a state occurrence, would give better results. In fact the results of the F-method and C-method are almost identical.

frame rate (ms)	states per model	Clustering Parameters occ.	Clustering Parameters Δ L	Av. speech per state (ms)	Clust. states per model posn.	P rank
10	3	25	50	561	2139	10
6	3	25	50	404	2966	4
6	3	42	0	573	2092	3
6	3	42	84	591	2030	1
6	3	42	168	883	1358	5
6	3	64	128	886	1354	2
6	3	128	256	1879	638	6
6	5	1	1	92	7843	12
6	5	5	10	143	5042	11
6	5	25	25	348	2066	8
6	5	25	50	350	2058	9
6	5	42	84	534	1348	7

Table 1. Columns 1-3 give the control parameters used in each experiment, columns 4 & 5 give the resulting system statistics, and column 6 gives the rank order of the performance of the systems using the P-method.

The I-method does not use the Gaussian distribution associated with each state. Instead a probabilistic distance measure similar to the Itakura-Saito distance is used. The speech in a 25ms frame at the centre of each state occurrence is compared to the P-method LP vector, and the speech from the best frame used to calculate the final LP coefficients. The I-method substantially outperforms both the C-method and the F-method.

4. EXPERIMENTAL RESULTS

One hour of training data was recorded from a single speaker at a sampling rate of 16kHz. This data was then pre-emphasised, Hamming windowed with 25ms frames, and coded into MFCCs with energy, deltas and accelerations, to give a 39 dimensional feature vector. Originally 3 state models were used with a 10ms frame rate. The left to right nature of the models results in a minimum model duration of 30ms. Later 5 state models were introduced in an effort to achieve better time resolution. To keep the minimum model duration at 30ms, the data was re-coded with a 6ms frame rate. A 20th order LP filter was used to characterise each clustered state. The dictionary used was a 45 symbol phone set development version of the British English Example Pronunciations dictionary currently being constructed at CUED.

The details of the different system configurations tested are given in Table 1. The first three columns of the table refer to the control parameters of each experiment. These are the frame rate in milliseconds, the number of states per model, and the minimum occupation and minimum change in log-likelihood clustering parameters. The fourth column gives the average amount of speech (in milliseconds) pooled into each clustered state. Note that the minimum amount of speech is given by the minimum occupation clustering parameter multiplied by the frame rate. The fifth column gives the average number of clustered states per model position. Finally the sixth column gives an informal impression of the rank order of the quality of the P-method results.

System	Error rate (%)
Natural speech	0.7
Resynthesised natural speech	3.3
rank-1 P-method synthesis	33.0
rank-7 P-method synthesis	34.0
rank-1 I-method synthesis	37.3
DECtalk 1.8, Paul	3.25
MITalk-79	7.00
Amiga SoftVoice	12.25
Votrax Type'n'Talk	27.44
Street Electronics Echo	35.56

Table 2. Modified Rhyme Test results for various synthesis systems and natural speech, including results from [10] for comparison.

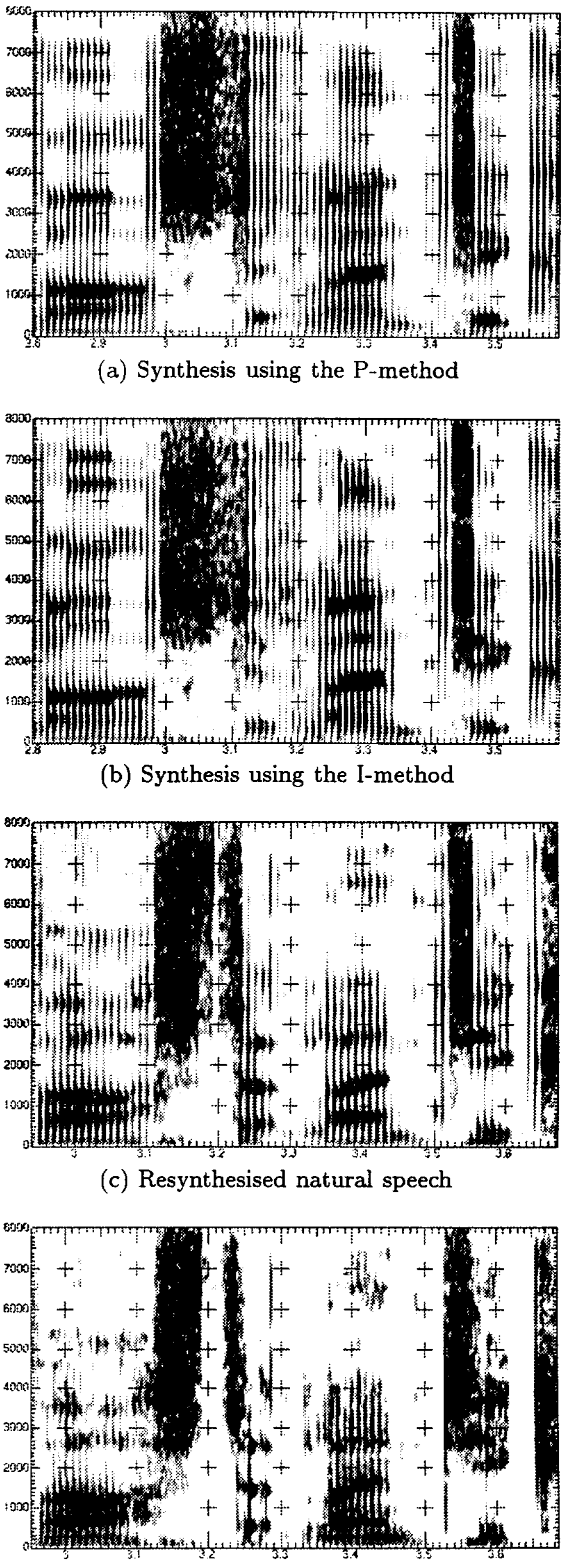
The two configurations ranked bottom in Table 1 performed considerably less well than the others. This was due to there being an insufficient amount of speech associated with each state to properly estimate a Gaussian distribution. The audible differences between the other configurations in the table were small, and the rank order assigned to them is not very precise. Configurations which sounded smoother, with less artifacts, were ranked higher. The best results were achieved using 3 state models with 6ms frame rate data. Better performance was obtained by having at least an average of 500ms of speech per clustered state, and less than 1,000 clustered states per model position seemed to be detrimental to performance.

Closed response Modified Rhyme Tests (MRT) were carried out using the methods described in [10], and the wordlists given in [11]. Only six different subjects were used to evaluate each type of speech in the current preliminary tests, compared with seventy-two in [10]. Tests were conducted to evaluate the performance of both the P-method and I-method versions of the top ranked configuration in the 3-states-per-model group in Table 1, and the P-method version of the top ranked configuration in the 5-states-per-model group. Tests were also carried out for natural speech, and resynthesised natural speech. The latter was obtained using standard autocorrelation method LPC techniques over 25ms frames at a 6ms frame rate, resynthesising on a monotone, and is the best that the current system can hope to achieve. Table 2 presents the results from the current tests, and also some of the results from [10] for comparison. Further analysis of the test results revealed that over 70% of the errors in the synthesised speech tested were due to poorly synthesised plosives.

To illustrate the speech generated by the system, Figure 1 shows wideband spectrograms of the sentence fragment "vast Atlantic", taken from the synthesised utterance "When a sailor in a small craft faces the might of the vast Atlantic Ocean today." Figure 1a used the P-method, and Figure 1b the I-method. The synthesised speech in both spectrograms was generated using the best system configuration in Table 1. Figure 1c is resynthesised natural speech, generated as described above, and is included for comparison as it is the best that the current system can hope to achieve. Figure 1d is natural speech.

5. DISCUSSION

HMMs assume that the speech signal can be modelled as piece-wise constant. The effect of this assumption on the synthesised speech can be seen in Figure 1. Here the formants remain constant for up to seven pitch pulses at a time, after which they jump to another constant position.



(a) Synthesis using the P-method

(b) Synthesis using the I-method

(c) Resynthesised natural speech

(d) Natural speech

Figure 1. Wideband spectrograms of the sentence fragment "vast Atlantic".

Such jumps could cause glitches in the synthesised speech, but with current systems none can be heard. Smoothing out these jumps makes a barely perceptible change to the speech quality. The context dependent structure of the models seems to have ensured that the formant jumps are so small that they are not noticeable.

The overall sound of the synthesised speech when synthesising sentence length material is very similar to that of resynthesised natural speech. The durations are particularly good, as can be seen from Figure 1. The MRTs revealed that the system has problems in synthesising plosives, and this can also be seen in Figure 1. The final /t/ of "Atlantic" is synthesised well. However the /k/ of "Atlantic" suffers from a voicing error in its middle state, and the first /t/ of "Atlantic" and the /t/ of "vast" do not contain the required closures.

Examination of the database has revealed two phenomena likely to be largely responsible for the poor performance of the system when synthesising plosives. Firstly, it was discovered that some plosives specified in the phone level transcriptions of each sentence in the database were either partially or completely missing from the speech of the database. This was due to imprecise speech (dropping of final plosives) on the part of the speaker used to record the database. The missing plosive triphones are therefore aligned to some other piece of speech, which is then included in the pool used to estimate the synthesis parameters. The Gaussians representing the states of these plosives are also poorly estimated, which leads to poor model alignment, and so further degradation of the synthesis parameters. Secondly, occurrences of both the unreleased and released forms of a plosive were found in the same phonetic context. The current system attempts to model both with the same triphone, which is inappropriate. Using a more precise, more consistent speaker could reduce the number of occurrences of both phenomena, though the latter would still be likely to occur in triphones present with differing wider contexts. However, one of the chief merits of this system is that it can be retrained on any voice, and so to restrict this capability in any way is undesirable. Hence alternative solutions involving altering the pronunciation dictionary and/or the individual model structures will be sought instead.

Other factors may also contribute to the poor synthesis of plosives. Examination of the database revealed that the closure in a plosive lasts for 30-70ms. When present, the following release takes the form of either a burst of turbulent noise lasting 20-60ms, followed by a further 10-30ms of /h/ like aspirated noise in the case of voiceless plosives, (/t/, /k/ & /p/), or of a burst lasting 15-40ms followed by formant transitions into the adjacent vowel in the case of voiced plosives, (/b/, /d/ & /g/). Given these short timescales, and the burst onset of the release, the treatment of the speech signal as stationary 25ms frames undoubtably has some effect on the quality of the synthesis parameters. The framing effect on its own leads to only a small degradation in the quality of synthesised plosives, as can be seen by the MRT score for resynthesised natural speech. However it is possible that interaction with the averaging that takes place in constructing the Gaussians of the HMMs, leading to poorly estimated Gaussians and hence poor model alignments, could be amplifying the effect. The possibility that plosive structure is too complex to be properly modelled by a three state model was addressed by the creation of five state models. However, considering all the problems discussed above, this was probably premature, and the resulting system performed no better in the MRTs. Finally, the use of single Gaussian distributions instead of Gaussian mixture distributions may be over optimistic.

The synthesised speech also suffers from the LP buzz effect. This buzz is particularly pronounced in the P-method because the LP coefficients tend to be blurred by the pooling of dissimilar speech. The formant bandwidths are therefore too wide, and hence the synthesised speech overdamped. The I-method does not pool speech, and so suffers less from the buzz effect. However it does suffer from occasional artifacts in the synthesised speech caused by either unrepresentative speech being used to estimate the LP coefficients for a state, or by the speech synthesised from a state sounding different to the database segment that it was based upon.

6. CONCLUSION

A new technique which uses HMMs to automatically estimate synthesis parameters from a speech database has been introduced. The resulting synthesis system produces intelligible speech which mimics the voice of the speaker used in the database.

Future work will be directed towards improving the segmental intelligibility of the synthesised speech and incorporating the PSOLA synthesis technique into the system.

ACKNOWLEDGEMENT

R.E.Donovan is funded by an EPSRC research studentship.

REFERENCES

[1] A. Falaschi, M. Giustiniani & M. Verola, (1989). *A Hidden Markov Model Approach to Speech Synthesis*, Proc. Eurospeech'89, pp. 187-190, Paris.

[2] E.P. Farges & M.A. Clements, (1988). *An Analysis-Synthesis Hidden Markov Model of Speech*, Proc. ICASSP'88, Vol. 5, pp. 323-326, New York.

[3] M. Giustiniani & P. Pierucci, (1991). *Phonetic Ergodic HMM For Speech Synthesis*, Proc. Eurospeech'91, Vol. 1, pp. 349-352, Genova.

[4] P.C. Woodland, J.J. Odell, V. Valtchev & S.J. Young, (1994). *Large Vocabulary Continuous Speech Recognition Using HTK*, Proc. ICASSP'94, Vol 2, pp. 125-128, Adelaide.

[5] S.J. Young, J.J. Odell & P.C. Woodland, (1994). *Tree Based State Tying for High Accuracy Acoustic Modelling*, Proc. ARPA Workshop on Human Language Technology, Merril Lynch Conference Centre.

[6] I.R. Murray & J.L. Arnott, (1993). *A Tool For The Rapid Development Of New Synthetic Voice Personalities*, Proc. ESCA Workshop on Speech and Language Technology for Disabled Persons, KTH, Stockholm, Sweden, pp. 111-114.

[7] F.J. Charpentier & M.G. Stella, (1986). *Diphone Synthesis using an Overlap-Add Technique For Speech Waveforms Concatenation*, Proc. ICASSP'86, pp. 2015-2018, Tokyo.

[8] J.L. Courbon & F. Emerard, (1982). *SPARTE: A Text-To-Speech Machine Using Synthesis By Diphones*, Proc. ICASSP'82, pp. 1597-1600, Paris.

[9] E. Moulines & F.J. Charpentier, (1990). *Pitch-Synchronous Waveform Processing Techniques for Text-to-Speech Synthesis using Diphones*, Speech Comm., Vol 9, No. 5/6, pp. 453-467.

[10] J.S. Logan, B.G. Greene & D.B.Pisoni, (1989). *Segmental intelligibility of synthetic speech produced by rule*, J. Acoust. Soc. Am. 86 (2), pp. 566-581.

[11] A.S. House, C.E. Williams, M.H.L. Hecker & K.D. Kryter, (1965). *Articulation-Testing Methods: Consonantal Differentiation with a Closed-Response Set*, J. Acoust. Soc. Am. 37 (1), pp. 158-166.