Proceedings of the Institute of Acoustics

ARTICULATORY COPY SYNTHESIS USING MULTIPLE CODEBOOKS

A.R. Greenwood (1), C.C. Goodyear (2)

(1) Now at Liverpool John Moores University, School of Electrical Engineering and Electronics, Byrom St., Liverpool
(2) Liverpool University, Department of Electrical Engineering and Electronics, Brownlow Hill, Liverpool

1. INTRODUCTION

A time domain synthesiser has been developed which uses nine parameters to control the shape of a model vocal tract. The model, similar to Mermelstein's, was designed with the aid of magnetic resonance imaging (MRI) to match the vocal tract of a chosen speaker. The parameters define the positions of the key articulators, from which the vocal tract outline is geometrically constructed. The area function is computed by superimposing a grid on the outline and using experimentally derived formulae relating sagittal width to cross-sectional area.

An acoustic-to-articulatory codebook was generated by randomly sampling the articulatory space spanned by the model. Copy synthesis was then performed by selecting a sequence of codebook entries with the aid of a dynamic programming algorithm, in conjunction with a cost function that took into account both spectral and geometric information.

The quality of the speech synthesised in this manner was restricted by the sparseness of the codebook. The use of a larger codebook is prohibited due to the storage requirements and the computational effort needed to search it. Clustering the codebook into regions in acoustic space, each defined by a voiced diphone, allowed smaller codebooks to be used. Small scale expansion of these sub-codebooks permitted a series of small densely populated codebooks to be generated. A complete utterance was then synthesised by using a different sub-codebook for each segment. Informal listening tests have revealed that speech quality was improved when synthesised using this method.

2. MAGNETIC RESONANCE IMAGING AND ARTICULATORY MODELLING

The use of MRI to obtain area data for use in articulatory synthesis is reported in [1]. Area functions were obtained for five different vowels, which when used to drive a Kelly-Lochbaum model [2], produced sounds with similar spectral properties to the natural vowels.

In order to perform spectral copy synthesis it is necessary to have a large database of vocal tract shapes. Schroeter [3] demonstrated that random sampling of the articulatory space defined by a Mermelstein [4] model, produced suitable entries for populating an acoustic-to-articulatory codebook.

The MRI data were used to develop a nine parameter model of the vocal tract similar to Mermelstein's. The mid-sagittal images were used to produce a stylised vocal tract outline from the positions of the main articulators, and the axial images were used to find a relationship between the sagittal width and the cross-sectional area in various regions of the tract.

An example of the model generated vocal tract outline for the vowel /i/ is shown in figure 1. A grid structure was superimposed on the outline and sagittal widths at various points along the vocal tract were obtained. These were converted to cross-sectional areas using the formulae obtained from the axial images, and this non-uniformly

sampled area function was then resampled at a constant spacing in order to obtain areas suitable for use in the 21 section synthesis filter.

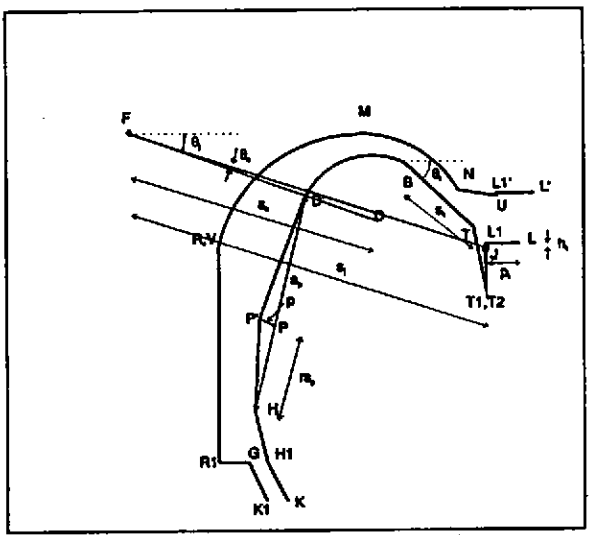

Figure1. Model generated vocal tract outline

3. ARTICULATORY SYNTHESIS BY CODEBOOK LOOKUP

In order to obtain high quality synthetic speech using an articulatory synthesiser, it is necessary to obtain a sequence of area functions that are a good approximation to those actually used by a human speaker. A method that has proven to be successful, is to look up suitable vocal tract configurations from codebook accessed using spectral information [3].

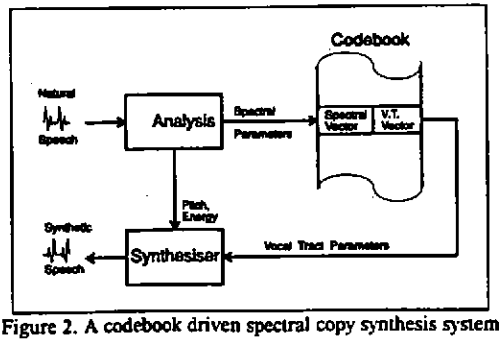

Figure 2. A codebook driven spectral copy synthesis system

A block diagram of a codebook driven copy synthesis system is shown in figure 2. The codebook consists of many entries, each of which consists of two components: a spectral vector and an articulatory vector. The spectral vector is obtained from the impulse response of a vocal tract filter, configured with the parameters given by the articulatory vector.

The natural speech waveform is analysed pitch synchronously, using a window three pitch periods long and positioned close to an excitation point, to gain a spectral representation of the centre pitch frame. This spectral vector is used along with a given codebook access function in order to obtain a set of articulatory parameters for use with the synthesiser, which also requires pitch and energy information computed directly from the speech waveform.

3.1 CODEBOOK GENERATION

A codebook entry was generated by assigning a random number, uniformly distributed within a defined range, to each parameter. The first 200 samples of the impulse response of the corresponding vocal tract filter were computed, and 12th order LPC analysis was performed. This allowed 14 cepstral coefficients to be calculated using a standard recursive formula [5], and the first three formant frequencies to be estimated from the roots of the predictor polynomial.

The codebook was pruned in two stages. First of all any physiologically impossible vocal tract shapes were removed, and in this way a codebook with 160703 entries was reduced to 91432 entries. Secondly when pairs of geometrically similar entries occurred, one was rejected. The formant space was divided into a set of three dimensional bins, and the length of each bin in any dimension was set to 5% of the starting frequency. The starting frequency along the F1,F2 and F3 axis were 150 Hz, 300 Hz, and 800 Hz respectively, while the numbers of bins in each dimension were 44, 48 and 33 respectively, as in [3]. The codebook entries were sorted into these bins, and if more than one entry lay in the same bin, and the geometric distance between them was below a predefined threshold, the entry furthest from the centre of the bin was discarded. The geometric distance measure is given by:

$$d_{bin} = \sum_{i=1}^{9} \left(p_i^{(1)} - p_i^{(2)} \right)^2 \qquad (1)$$

where $\left\{ p_i^{(j)} \right\}$ are the model parameters corresponding to the entry j.

The nature of the final codebook depends on the choice of the threshold; if a large threshold is used then a small, sparse codebook will be generated, and if a small threshold is used then the final codebook may be too large for practical use. A threshold of 1.5 was used to produce a final codebook size of 23780.

Figure 3 shows typical vowel and semivowel formant positions used by the chosen speaker, the vowels were found to lie within the regions found experimentally by Peterson and Barney [6]. Figure 4 shows the formant space covered by our codebook, where the large dots correspond to densely populated regions. It can be seen that some areas are sparsely populated, such as those occupied by the vowel /i/ and the start of the glide /j/, and there is a large number of entries in regions not used by our speaker.

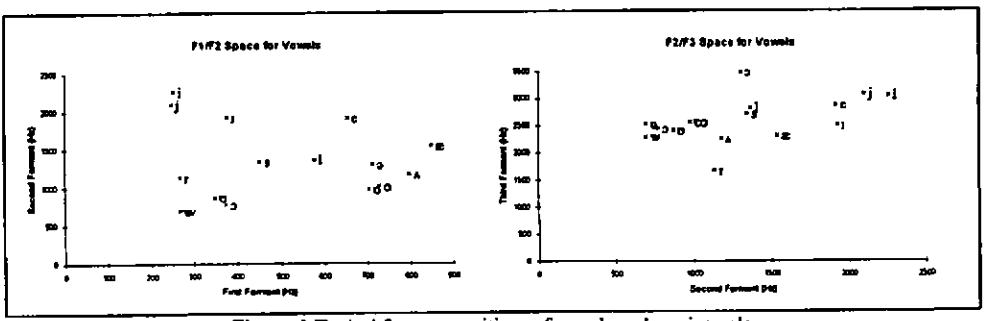

Figure. 3 Typical formant positions of vowels and semivowels

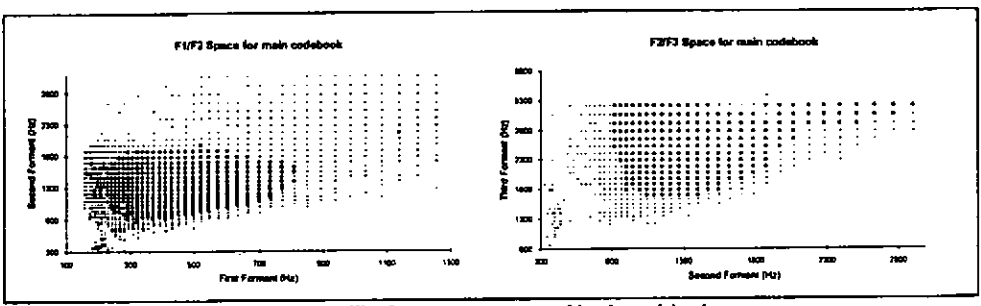

Figure 4. The formant space covered by the codebook

3.2 CODEBOOK ACCESS

The cost function used to access the codebook consisted of two components: a cepstral distance measure to produce a good spectral match, and a geometric distance measure to penalise large changes in area function. In order to reduce the influence of spectral tilt, a bandpass lifter was used, and the effect of the higher frequency formants was reduced by convolving the cepstra with $1+z^{-1}$. Therefore the filtered liftered cepstral distance measure is given by:

$$d_{ceps}(\mathbf{S},\hat{\mathbf{S}}) = w_1^2(c_1-\hat{c}_1)^2 + \sum_{k=2}^{14} w_k^2\left[(c_k+c_{k-1})-(\hat{c}_k+\hat{c}_{k-1})\right]^2 \qquad (2)$$

where S is the cepstral vector derived from the natural speech with coefficients $\{c_k\}$, $\hat{\mathbf{S}}$ is the cepstral vector derived from the synthetic speech with coefficients $\{\hat{c}_k\}$, and $\{w_k\}$ is the bandpass lifter suggested by Juang *et al* [7], and given by:

$$w_k = \frac{1+7\sin(k\pi/14)}{8} \qquad (3)$$

The geometric distance measure preferred here is given by:

$$d_{geo}(\mathbf{A}_n, \mathbf{A}_{n-1}) = \sum_{i=1}^{21} \left(\ln\left(A_i^{(n)}\right) - \ln\left(A_i^{(n-1)}\right) \right)^2 \quad (4)$$

where $\mathbf{A}_n = \left(A_1^{(n)}, A_2^{(n)}, \ldots, A_{21}^{(n)}\right)$ is the area vector corresponding to fame n.

Therefore the total cost function is given by:

$$C = d_{ceps}\left(\mathbf{S}_n, \hat{\mathbf{S}}_n\right) + w_{geo} d_{geo}(\mathbf{A}_n, \mathbf{A}_{n-1}) \quad (5)$$

where w_{geo} is an empirically determined constant. The codebook is then searched for each frame of natural speech in turn, and the entry that minimises the cost function is used to drive the synthesiser.

The quality of the synthetic speech was assessed using two different distance measures: a log spectral distance measure d_{ls}, and smoothness measure d_{sm}, to determine how much the area function is changing between frames. These distance measures are given by:

$$d_{ls} = \frac{1}{N_F} \sum_{j=1}^{N_F} \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left[\left(P_i^{(j)} - \overline{P}^{(j)}\right) - \left(\hat{P}_i^{(j)} - \overline{\hat{P}}^{(j)}\right) \right]^2} \quad (6)$$

$$d_{sm} = \sqrt{\frac{1}{21 N_F} \sum_{j=2}^{N_F} \sum_{i=1}^{21} \left(\frac{A_i^{(j)} - A_i^{(j-1)}}{A_i^{(j-1)}} \right)^2}$$

where $P_i^{(j)}$ and $\hat{P}_i^{(j)}$ are the i'th frequency samples of the natural and synthetic speech spectra for frame j in dB respectively, $\overline{P}^{(j)}$ and $\overline{\hat{P}}^{(j)}$ are the mean values of the natural and synthetic speech spectra, $A_i^{(j)}$ is the i'th area from the j'th frame, and N_F is the number of voiced frames.

For the utterance "We were away a year," when w_{geo} was set to zero, the spectral distortion was 4.29 dB and the smoothness was 23.6. When w_{geo} is increased to 0.1, then the spectral distortion falls to 4.10 dB, and more significantly the smoothness reduces to 3.2.

3.3 DYNAMIC PROGRAMMING

More natural sounding speech can be produced by applying a dynamic programming algorithm [8]. If T frames of natural speech $\mathbf{S}_n$, n=1,2,...T, are analysed at a time; then the problem is to select a series of area functions $\mathbf{A}_n$, n=1,2,..T, which produces synthetic speech frames $\hat{\mathbf{S}}_n$, $n = 1.2,\ldots,T$, in order to minimise the cumulative cost function given by:

$$d_{dp}(T) = d_{copy}\left(S_1, \hat{S}_1\right) + \sum_{t=2}^{T}\left[d_{copy}\left(S_t, \hat{S}_t\right) + w_{proc}\, d_{proc}\left(A_t, A_{t-1}\right)\right] \qquad (7)$$

If the codebook contains M entries, the it can be thought of as selecting the optimum path through a MxT matrix as shown in figure 5. Although there are M^T possible paths through the trellis, only one path to each node in any column needs to be retained, and the computational cost can be reduced to M^2T.

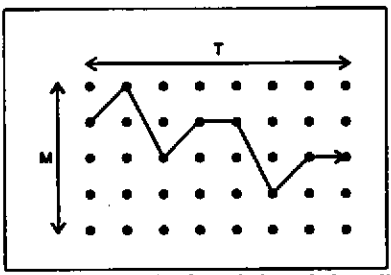

Figure 5. An example of a path through the trellis

In order to reduce the computational load, each column of the trellis was filled with 100 entries selected by comparing the stored spectral vector to the current speech frame. With T set to 15 and w_{proc} to 0.01, the spectral distortion was 4.02 dB and the smoothness was 1.1.

4 DIPHONE SYNTHESIS

The main disadvantages with using a large codebook generated by the random sampling method, are that large areas of acoustic space are sparsely covered, while some regions that are never used have a large number of entries. A more evenly spaced codebook can be obtained if a large number of random samples are taken, for instance Schroeter and his colleagues use 65 million samples to generate a codebook with 160809 entries [3], however this approach is prohibited when using a 25 MHz i486 personal computer, and still has the disadvantage that a large number of entries are never used.

An approach to reducing the need for huge computer effort is to identify the regions of the acoustic space that are actually used by the speaker, and to populate these areas densely. In order to accomplish this it was decided to divide the acoustic space into several zones, each defined by a voiced diphone.

A diphone (diad) is defined as being the segment from the stationary part of one phoneme to the stationary portion of the next phoneme. As there approximately 40 phonemes in English there should be about 1600 diphones, some of which will not occur, but by taking into account allophones and the difference between stressed and unstressed syllables, Wang and Peterson [9] estimated that there are about 8000 different diphones in any dialect of English, however practical diphone based text-to-speech systems cope with an inventory of about 1000 [10].

In order to test the feasibility of this approach, it was decided to concentrate on the single utterance "We were away a year". This can be divided into the diphone units: /(wi) (iw) (wə) (ər) (rə) (əw) (weɪ) (ɪə) (əj) (jɪə)/, and can be synthesised from an inventory of six diphones: /wi/, /əw/, /ər/, /weɪ/, /ɪə/ and /jɪə/, referred to WE, AW, ER,

WAY, IR and YEAR in the text. The diphone /əj/ is generated from the diphone YEAR because both target positions are contained in this codebook.

In order to generate these sub-codebooks, recordings of the chosen speaker continuously voicing the required diphones were made. These were then analysed a frame at a time and for each frame the 50 best spectral matches from the large codebook were obtained. Formant tracks were calculated for the diphones using the algorithm suggested by Markel and Gray [5], allowing the region in acoustic space which contains the diphone to be identified. Any sub-codebook entries not in this zone were eliminated.

The next stage was to remove any entries without a close geometric neighbour; each sub-codebook was clustered into four clusters using the modified k-means clustering algorithm [11], on the basis of a spectral distance measure. The geometric centre of each cluster was identified, and the mean and standard deviation of the distance from the centroid to all other entries was computed. Entries more than four standard deviations above the mean were removed, in such a way that no more than 5% of a cluster was pruned. The size of each sub-book before and after pruning is shown in table 1.

Table 1. Sub-codebook sizes before and after pruning

Sub-codebook	Original Size	Size before spectral pruning	Size after geometric pruning
WE	1003	717	686
AW	810	617	599
ER	1103	1020	986
WAY	2559	2395	2292
YEAR	1575	1465	1357
IR	745	597	570

Synthesis of the chosen utterance using dynamic programming within these sub-codebooks resulted in a spectral distortion of 4.02 dB, and a decrease in the smoothness measure to 0.7.

In order to increase the density of the codebook, nine additional entries were generated for every original entry, by adding a small random number to each parameter. Synthesis with these new codebooks resulted in a spectral distortion of 3.96 dB, and a smoothness factor of 0.6. Informal listening tests revealed that the speech sounded smoother.

5 DISCUSSION

A nine parameter model of the vocal tract has been developed from magnetic resonance images. Random sampling of the articulatory space has created a codebook which can be used for spectral copy synthesis. The use of a filtered liftered cepstral distance measure, along with a Euclidean geometric distance penalty over successive frames has produced intelligible speech, while the extension of this technique using a dynamic programming algorithm has lead to much improved speech with smoother area changes.

The acoustic space has been divided into regions, defined by given voiced diphones. Entries have been extracted from the original codebook which lie in these regions of acoustic space, and entries geometrically very different from the rest have been pruned. This has lead to smoother sounding speech, which can be further improved if each of the codebooks is expanded by adding a slight jitter to the parameters of the original codebook entries.

It is known that multi-layered perceptrons (MLPs) can be used to perform acoustic-to-articulatory mappings of small portions of the acoustic space [12], obtained using a simple clustering algorithm. This has the disadvantage that several MLPs were trained on data that was never used. The pruning and expansion stages described here should result in a better set of exemplars for training purposes.

6 ACKNOWLEDGEMENTS

The authors are grateful to Dr. P.A. Martin or the magnetic resonance imaging centre for obtaining the images, to Prof. R.H.T. Edwards for extending to us the use of the MR facility, and to Dr. M.C. Hall or British Telecom Laboratories for his financial assistant under the CASE award scheme administered by the SERC.

7 REFERENCES

[1] Greenwood A.R., Goodyear C.C. Martin P.A., "Measurements of Vocal Tract Shapes Using Magnetic m, Resonance Imaging," Proc. IEE Part I, to be published, Dec 1992.

[2] Kelly J.L., Lochbaum C.C. "Speech Synthesis," 4th Int. Cong. Acoustics, Copenhagen, Denmark, pp.1-4, 1962.

[3] Schroeter J., Meyer P., Parthasarathy S. "Evaluation of Improved Articulatory Codebooks and Codebook Access Distance Measures," Proc. IEEE Conf. Speech Signal Proc., Albuquerque, USA, pp. 393-396, 1990.

[4] Mermelstein P. "Articulatory Model for the Study of Speech Production," J. Acoust. Soc. Am., Vol. 53(4), pp.1070-1082, 1973.

[5] Markel J.D., Gray A.H. "Linear Prediction of Speech," Springer Verlag, New York, 1976.

[6] Peterson G.E, Barney H.L. "Control Methods Used in a Study of the Vowels," J. Acoust. Soc. Am., Vol. 24(2), pp. 175-184, 1952.

[7] Juang B.H., Rabiner L.R., Wilpon J.G. "On the Use of Bandpass Liftering in Speech Recognition," IEEE Trans. Accost. Speech Signal Proc., Vol. ASSP-35(7), pp. 947-954, 1987.

[8] Schroeter J. , Sondhi M.M. "Dynamic Programming Search of Articulatory Codebooks," Proc. IEEE Conf. Speech Signal Proc., Glasgow, Scotland, pp. 588-591, 1989.

[9] Wang W. S-Y., Peterson G.E. "Segment Inventory for Speech Synthesis," J. Acoust. Soc. Am., Vol. 30(8), pp.743-746, 1958.

[10] Klatt D. "A Review of Text-to-Speech Conversion for English," J. Acoust. Soc. Am., Vol. 82(2), pp. 737-793, 1987.

[11] Wilpon J.G., Rabiner L.R. "A Modified K-Means Clustering Algorithm for Use in Isolated Word Recognition," IEEE Trans. Acoust. Speech Signal Proc., Vol. ASSP-33(3), pp. 587-595, 1985.

[12] Rahim M.G. "Neural Networks in Articulatory Synthesis," Ph.D. Thesis, University of Liverpool, 1991.