A Hidden Markov Model Approach to Speech Synthesis

Alessandro Falaschi*, Massimo Giustiniani**, Massimo Verola*

* La Sapienza Univ. of Rome, INFOCOM Dept. Via Eudossiana 18, 00184 Roma, Italy
** IBM Rome Scientific Center, Via del Giorgio 159, 00143 Roma, Italy

ABSTRACT

This paper describes a novel technique for producing smooth speech parametric representation evolution by means of an application of traditional hidden Markov modeling techniques. It is based on the consideration that HMM training is able to locate the main acoustic events which occur during the speech process; thus the obtained decomposition can be used to reproduce the linguistic units utilized for training. This is accomplished by interpolating the state-related features values with some weighting functions, as it is done for temporal decomposition technique [1] [15]. Results will be given for isolated word synthesis.

I - INTRODUCTION

Hidden Markov models have extensively been utilized in speech recognition systems [2] [3] [4], as they provide a robust representation of the main events in which speech can be segmented; moreover, the existence of efficient HMM parameters estimation techniques allows an automatic model identification.

The preliminary experiments here presented refers about synthesis of whole linguistic units, as words. Although this could be also regarded as a simple speech coding process, this is not the case because of the averaging properties of the training process. The resulting synthesized speech quality will result unrelated from a particular speaker, or from an a priori parameter choice, thus giving to the synthetic speech some objectivity properties. In particular, the estimated models will represent the common characteristics of the whole speech material utilized for training, as it has been proved for speaker-independent recognition systems [4], [16] and multi-style training methods [7].

Before to give the particulars about the synthesis process and the definition of the weighting functions utilized for interpolating the feature vectors related to the states of the HMM utilized for synthesis, we introduce a new method for inferring the acoustical structure of linguistic events, which is based on the utilization of an ergodic hidden Markov model (EHMM) of speech.

The Viterbi algorithm, applied to the EHMM, gives a segmentation of a sample utterance in terms of its elementary acoustical units; the obtained segmentation gives a suitable initialization of the HMM that will be used for synthesis. This is performed by utilizing either the initial model or the result of the Baum re-estimation procedure. Different choices of the weighting functions are experimented, and the results given in form of spectrogram.

II - THE ERGODIC HMM

An ergodic HMM is an hidden Markov model whose transition matrix is completely connected, and whose states have no linguistic identity. Once the EHMM is trained over a large sample of speech, the spectra associated to the states will represent the main acoustical events found, and the transition matrix the phonotactical constraints of the language.

For the present experiments we have utilized a 64 states EHMM, characterized by continuous observation densities belonging to the autoregressive gaussian family [8], defined by a set of P prediction coefficients \( a_k \) for state \( S_k \). Each speech frame, constituted by the speech signal samples \( x_n, n = 1 \ldots M \), is represented by means of its normalized autocorrelation:

\[
\gamma^2 = \frac{N}{\sigma^2} \sum_{n=1}^{M} x_n^2
\]

where \( \sigma^2 \) is the total prediction error related to an LPC analysis performed on the frame, and \( N \) is a constant experimentally set to 10. The probability of observing vector \( R = \{ r_1 \} \) while staying in state \( S_k \) has the expression [9]:

\[
P_k(R) = \frac{N}{2\pi} e^{-\frac{1}{2} \delta (R, a^k)}
\]

where

\[
\delta (R, a^k) = r_0^k x + \sum_{i=1}^{p} r_i^k x_i
\]

in which \( r_k^a \) is the autocorrelation function of the P prediction coefficients. The so-obtained autoregressive density lies in between the gain-independent one utilized in [10], the one derived in [17], and the density utilized by Poritz in [8].

The prediction order utilized into the experiments is set to 14, with a frame rate of 8 msec and a frame length of 320 samples at the sampling rate of 10 KHz. The LPC analysis is preceded by an hamming windowing of the preemphasized speech frame.

The state-related spectra is initialized by a Lloyd Vector Quantization algorithm [11] operated over about eight minutes of speech; the same material is then utilized to run the Baum [6] algorithm in order to obtain the EHMM parameters maximum likelihood estimate.
III - EHMM FILTERING AND DECODING

As a by-product of the Baum algorithm, the analysis of an utterance gives for each instant $h$ and each of the states $S_k$ of the EHMM a probability $P_k(h)$ of being in state $S_k$ while observing frame $h$. Such probability values can be adopted for utilization of the EHMM as a filter, estimating the filtered observation autocorrelation vector $r'(h)$ at frame $h$ as its expected value with respect to the states of the model:

$$r'(h) = \mathbb{E}_k \{ r(h) \} = \sum_k P_k(h) r(h)$$

from which the frame-related LPC filtered spectra can be recovered.

Fig. 1 reports the spectrogram of the word 'parole' (words), together with its filtered version and the shape of the interpolating function $P_k(h)$. Fig 2a shows the spectrogram obtained by utilizing, for each frame $h$, the LPC spectra related to the state $k^*$ for which

$$k^* = \arg \max_k \{ P_k(h) \}$$

A similar drawing can be obtained in Fig. 2b by means of the Viterbi algorithm, which gives the most likely states path among the EHMM while observing the actual speech of Fig. 1, obtaining a sort of acoustic decoding.

IV - HMM SYNTHESIS

Before to continue the exposition, let us have a look to the synthesizer structure shown in Fig. 3: each frame of speech is parametrized by a vocal tract model, obtained from a set of Log Area Ratios (LAR) coefficients derived from an LPC analysis, a Voiced/Unvoiced flag (V/UV), a Log Energy (LE) value and a Pitch Period (PP) value. This feature vector, which will be indicated as $F(h)$, completely defines the characteristics of the $h$-th frame of speech and can correctly drive the synthesizer.

The aim of the work is to derive the $F(h)$ sequence by utilizing the parameters which identify the HMM with the left-to-right structure shown in Fig. 4. The number of states are derived by means of the Viterbi algorithm by decoding, via the EHMM, a single occurrence of the training word. Such states are associated to a joint observation density whose components are the features needed by the synthesizer. Assuming, as an acceptable approximation, the statistically independence of the parameters, we can compute the joint density as the product of the marginal component densities. The autoregressive density is defined as for the EHMM and its mean value is initialized by the Viterbi decoding; LE and PP are described by two gaussian densities, and the V/UV flag by a discrete random variable. Let us denote the mean value of the joint feature vector density for state $S_k$ as $M_k$.

Smooth evolution of the $F(h)$ feature vector have been experimentally obtained by substituting the probability functions $P_k(h)$ in (4) by some other weight function $W_k(h)$, thus obtaining

$$F(h) = \frac{\sum_k W_k(h) M_k}{\sum_k W_k(h)}$$
We have experimented two choices for the $W_k(h)$: the first one derive from a particular density which has be taken in [12] as a realistic duration density for speech events, which has the expression

$$W(n) = 2 e^{-\frac{\mu k^2}{2} \sinh \left( \frac{\mu k}{2} \right)}$$

The other choice for $W_k(h)$ is a more pragmatic trapezoidal weighting function, reminiscent of some temporal decomposition synthesis technique [13].

In order to correctly compute the $W_k(h)$ functions, we define for each state $S_k$ a mean duration time $D_k$ as

$$D_k = \frac{1}{1 - p_k}$$

where $p_k$ is the state loop probability. Function $W_k(h)$ begins at instant

$$h = \sum_{n=1}^{k-1} D_n$$

and its length depends on the $D_k$ value. The $\mu$ parameter of (7) is obtained by equating the function (7) mean value to the $D_k$ value via numerical methods, as described in [14].

Fig. 5 shows the shapes of the two weighting functions, taking as mean state duration time the segmentation obtained by the time alignment of Fig. 7.

V - HMM TRAINING

The HMM derived by the Viterbi alignment of the EHMM with a single occurrence of a word has been submitted to a Baum re-estimation algorithm in order to get a maximum likelihood estimation for its parameters value, thus retaining the common characteristic of all the speech utilized for training. At present, this has been done only for what regard the spectral density; the other features densities have been derived in a different way. By first, all the occurrences of the training word have been aligned with the re-estimated HMM on a spectral basis. Then, the feature vector components statistics are collected from such alignments, obtaining mean and variance estimates for the missing densities parameter values.

VI - RESULTS AND DISCUSSION

Fig. 6 shows the spectrogram of the synthesized word 'parole' by using as weight functions the duration model ones, and as HMM the one directly obtained via Viterbi alignment of the EHMM.

Fig. 7 shows the re-estimated HMM via the Baum algorithm, in a form comparable with the one of Fig. 2. Its parameter values are then used for obtaining the synthesized spectrogram of Fig. 8, utilizing the trapezoidal weight function.

Either the initial and re-estimated models, as well as the two kinds of weighting functions, seems well suited for reproducing an intelligible and natural speech quality when
used for synthesis purposes, on the basis of some informal listening.

The advantages of the illustrated method as a synthesis technique relies on its intrinsic data compression properties, which made it appealing on a memory storage economy basis. In fact, the 70 frame of speech which occur in the utterance of the sample word are synthesized by means of the parameters of only 13 states, thus obtaining a compression rate over 5:1. Moreover, in the case EHHMM alignment is used for building words HMMs, the EHHMM state parameters itself could be utilized for synthesis, thus obtaining a description of speech based on a finite alphabet of acoustic events, in a way similar to Vector Quantization techniques [18].

Finally, we wants to make a comparison among the illustrated techniques and the temporal decomposition one [13]. In that case, speech spectra evolution is reconstructed by interpolating a set of target spectra, utilizing a set on associated weight functions. When more than two weight functions overlap, the spectra related to the central one is tentatively interpreted as a strongly coarticulated articulatory target. By examination of Figg. 1c ad 5, it can be noted that the exposed technique has the same decomposition properties. Moreover, the flexibility of the hidden Markov modelling techniques allows to extend the principle to each of the parameters needed by the synthesizer schema of Fig. 4.

VII - FUTURE WORK

Although the exposition has been focused on speech synthesis, it will be very interesting to further explore the coding capabilities of the ergodic hidden Markov modeling technique [18].

In the field of speech synthesis, the natural continuation of the research will be to look for linguistic subword units HMMs, whose size and number choice permits to realize either a diphonic, demisyllabic or morphological synthesizer. Such synthesis techniques could be further improved by the Markovian approach by using, as an example, transition diagrams different from the one of Fig.4 and explicit inter-units transition matrices.

Finally, let us remark that the historical utilization of HMMs for recognition has not been lost. In fact, the HMMs of sub-word units trained for synthesis could indeed be used also for recognition tasks. An unified point of view on modeling both sides of the continuous speech communication process by means of HMM will undoubtedly improve both synthesizer quality and recognition accuracy.

REFERENCES