LOW DELAY PHONE RECOGNITION

José A. R. Fonollosa, Eloi Batlle and José B. Mariño
Dept. of Signal Theory and Communications, Universitat Politècnica de Catalunya.
c/ Jordi Girona 1-3. Edifici D5
Barcelona 08034, SPAIN
Tel: +34 93 4016440; fax: +34 93 4016447
e-mail: adrian@gps.tsc.ups.es

ABSTRACT

In the frame of the ACTS VIDAS project (VIDeo ASsisted with audio coding and representation) the authors have compared different schemes for low-delay phone recognition. In the VIDAS project, the phonetic recognizer was proposed for lip-synchronization in videophone applications, but the results are also of interest for very low rate speech coding (Phonetic Vocoding) and for driving synthetic mouth gestures in real time.

1 INTRODUCTION

Nowadays, phone recognizers are widely used in speech recognition systems. For example, in large-vocabulary applications, the recognized sequence of phones can be useful to select a first short list of words corresponding to some segment of speech, and then the detailed match score for each of these words is computed.

For this kind of applications, the research efforts have been directed to reduce the complexity, in terms of computational cost, of the recognition system. The accuracy of the estimated sequence of phones or the delay in obtaining the segmentation of the speech was not the principal concern since, in any case, we will have to wait until the end of the word or sentence to obtain the final word sequence with a more detailed search.

In this paper, we are interested in designing phone recognizers for specific applications as low-rate speech coding (Phonetic Vocoding) [1, 2], lip-synchronization in videophone coding [3], and real-time facial animation [4]. In these cases we have a bidirectional person to person communication and reducing the delay of any part of the communication system is always of interest for a fluent conversation.

Our efforts will be directed to reduce the delay of the phonetic recognizer without reducing the accuracy of the recognition results. Nevertheless, each application may require a different degree of accuracy and may have a different way of measuring the performance of the phone recognizer. In the VIDAS project, for example, the objective were not the phones by themselves but the estimation of the mouth visible articulatory parameters.

The paper is organized as follows: Section 2 is a short review of the standard HMM-based architecture for phone recognition that specifies the contribution of each stage to the global delay of the system. Section 3 presents different alternatives to reduce the delay introduced by the dynamic programming or decoding stage. Finally, Section 4 shows the results obtained with a grammar-free experiment on low-delay phone recognition using the SpeechDat [7] database.

2 HMM-BASED PHONE RECOGNIZERS

Hidden Markov modeling (HMM) and dynamic programming is still the usual choice for reliable speech recognition systems[5, 6]. In this section we define the delay of a phone recognizer and we shortly review the different sections of a HMM-based phone recognizer and their influence in that delay.

In this paper we will consider only the intrinsic delay of the selected recognition algorithm. In a real system, the computations associated with the implementation of the algorithms will introduce an additional delay that will depend on the speed of the processor. Nevertheless, phone recognition is an easy task for the last generation of microprocessors and the additional computational delay can be negligible.

The recognizer delay can be defined as:

2.1 Recognizer delay

The delay of a phone recognizer is T units of time if the classification and segmentation of a speech signal in phones up to time t is based only on the speech signal received up to time $t + T$.

The contribution of the different sections of a HMM-based phone recognizer to this delay is the following:

2.1.1 Framing

The speech signal is processed in frames. This segmentation is defined by the window duration (WS) and the window displacement (WD). As in any kind of block processing, the window duration is the cause of the first

[1]This research was supported by the CICYT Spanish research project TIC95-1022-C05-03

delay. In the following sections of the recognizer the delay will be measured in terms of frames units. The displacement of the window in time units gives the factor of conversion from frames to time units.

2.1.2 Feature Extraction

The computation of the basic parameters from the windowed speech frame is a memory-less process that causes no additional delay. Nevertheless, a good performance can not be obtained without adding time derivatives of the basic static parameters. This computation introduces a delay proportional to the number of frames considered in the regression formula. The number of previous and next frames used for the first (DF) and second derivative (AF) causes a total delay of DF+AF frames. Additionally, other time-domain filtering for channel adaptation purposes may introduce further delay in this parameterization stage.

2.1.3 Vector quantization

In recognizers based on Discrete Hidden Markov Models (DHMM) or Semi-Continuous Hidden Markov Models (SCHMM) the output vectors from the parameterization stage (including time derivatives) are classified by vector quantizers. This memoryless preprocessing simplifies the last stage of the recognizer (dynamic programming) without introducing any delay in the recognition.

2.1.4 Dynamic programming

The hidden Markov model is a statistical model that uses a finite number of states and the associated state transitions to jointly model the temporal and spectral variations of signals.

The Dynamic Programming stage uses the hidden Markov model of each phone and the sequence of observations given by the previous stage to determine the *optimum* segmentation in phones of the speech signal.

The one-pass (one-state) extension of the Viterbi algorithm is the usual choice to solve the "connected word (phone) recognition" problem [5]. As in the Viterbi algorithm, the standard one-pass algorithm computes best paths to every reference word (phone) state at every test frame and eventually is able to backtrack the best score to give the best word (phone) sequence.

In typical applications of the one-pass algorithm as connected digit recognition, the backtracking is usually performed at the end of the utterance after selecting the best final state. In the following section we discuss different backtracking strategies for real-time low-delay phone recognition and in Section 4 we show their performance in terms of phone deletions, substitutions, and insertions.

If we denote by DD the decoding delay introduced by the Dynamic Programming stage, we can finally express the total delay of the recognition system as

$$TD = WS/2 + WD(DF + AF + DD) \quad (1)$$

3 REAL-TIME BACKTRACKING

3.1 The one-pass DP algorithm

Using m to represent the frame index, $1 \leq m \leq M$, v to represent the phone model index, $1 \leq v \leq V$, and n to represent the state index of each model, $1 \leq n \leq N_v$, then for each frame the one-pas algorithm calculate the accumulated log likelihood, $d_A(m,n,v)$ as:

$$d_A(m,n,v) = d(m,n,v) + \max_{1\leq j\leq n}[d_A(m-1,j,v) + \log(a^v_{jn})] \quad (2)$$

For $1 \leq n \leq N_v$, $1 \leq v \leq V$, where $d(m,n,v)$ is the local log likelihood for state n of phone v and a^v_{jn} is the probability of transition from state j to state n. The algorithm also needs to update the starting frame of phone v in the optimum path passing by its state n, $b(n,v)$ as:

$$b(n,v) = b(\arg\max_{1\leq j\leq n}[d_A(m-1,j,v) + \log(a^v_{jn})], v) \quad (3)$$

The above recursions are carried out for all internal states of each phone model (i.e., $n > 2$). For the initial state of each phone, i.e. $n = 1$, we have the recursion

$$d_A(m,1,v) = d(m,1,v) + \max[D(m-1), d_A(m-1,1,v) + \log(a^v_{11})] \quad (4)$$

and if $D(m-1) > d_A(m-1,1,v) + \log(a^v_{11})$, i.e., there is a transition between phones, then the associated starting frame is updated

$$b(1,v) = m \quad (5)$$

In Equation (4) $D(m)$ is the maximum log likelihood of the last states of all the phone models

$$D(m) = \max_{1\leq v\leq V}[d_A(m,N_v,v) + \log(1 - a^v_{N_v N_v})] \quad (6)$$

Finally, we also need to save the transitions between templates for the backtracking step: $i(m)$ stores the index of best final phone

$$i(m) = \arg\max_{1\leq v\leq V}[d_A(m,N_v,v) + \log(1 - a^v_{N_v N_v})] \quad (7)$$

and $k(m)$ its starting frame

$$k(m) = b(i(m), N_{i(m)}) \quad (8)$$

At the end of the utterance, the final phone model of the best path is selected as

$$I = i(M) \quad (9)$$

$$K = k(M) \quad (10)$$

and the best phone sequence is determined iteratively using the backtracking pointers $i(m)$ and $k(m)$ [5].

3.2 Decoding delay

If we require a real-time phone recognizer with a limited decoding delay the termination Equation (9) and the associated backtracking has to be adapted in order to determine the best path at each frame iteration. If we consider a decoding delay of Δm frames, we will have to follow the backtracking pointers of the selected best path to determine the associated phone index Δm frames before.

For the determination of the best path we can consider the last state of each phone as in Equation (6) or all the phones of all the states, i.e.

$$\tilde{D}(m) = \max_{1 \leq v \leq V} \max_{1 \leq n \leq N_v} [d_A(m, n, v)] \qquad (11)$$

A priori, the selection of the best path using the last states of the phone models, i.e. Equation (6), seems the best way to obtain a good estimation of the phone sequence. Nevertheless, this estimation has the intrinsic delay associated with the (minimum) duration of the phone model. Our preliminary results confirmed clearly that Equation (11) provides best results, specially for very-low delays.

In both cases, the phones indexes associated with the last Δm frames of the utterance are still determined using the standard final Equation (9) and its corresponding backtracking.

3.3 Path Pruning

In general, the best path at frame m will be different from the best path selected at the end of the utterance. Nevertheles, the backtracking information at frame $m - \Delta m$ and previous frames tends to be the same if we choose a reasonable long delay. The results show that for delays of more than 100 ms, the performance of the real-time low-delay phone recognizer is equal to the performance of the standard single backtracking version.

However, in some applications a shorter delay can be of interest. In this case we observed that a reduction in the delay cause an important fast degradation in performance due to the frequent unrestricted jumps from one path to a complete different one.

When we reduce the decoding delay Δm, the selected backtracking index is likely to change from frame to frame, thus overestimating the number of phone segments. To avoid jumps from one path to a complete different path and the resulting insertions of phones, we included a pruning stage after the selection of the best path. At frame m, only the paths that share the best backtracking information at frame $m - \Delta m$ survive. Therefore, after the pruning stage, all the active paths share the backtracking information up to time $m - \Delta m$. Even if are continuously changing from one path to another, we will have now smooth transitions without erratic phone insertions.

4 RESULTS

Severel phone recognition experiments were conducted in order to evaluate the effect of reducing the delay in the real-time backtracking schemes without and with path pruning. The 26 context-independent phone models were trained using a total of 277844 phone realizations from 5341 sentences (805 speakers) of the SpeechDat (M) [7] Spanish Fixed Network Corpus. The HMM topology used is left-to-rigth HMM, with 4 states per phone and with all forward jumps allowed except the jump from state 1 to state 4. Hence, the minimum duration of a phone is 3 frames (30 ms).

The parameters that defines the different sections of the recognizer are the following:

Sampling Frequency: 8000 samples/s.

Window size (WS): 30 ms.

Window displacement (WD): 10 ms.

Parameterization: 14 MFCC coefficients with cepstral liftering and 20 bank filters

Delta frames (DF): 2

Acceleration frames (AF): 1

For testing, we selected a different set of 300 utterances from the same database. In the following subsections, we present the results of a grammar-free phone recognition experiment using. In both cases Equation (11) was used to select the best frame at each iteration

4.1 Low-delay phone recognition without path pruning

In this first experiment we do not introduce any pruning. Figure 1 shows the resulting number of insertions, deletions and substitutions as a function of the decoding delay in frames introduced by the dynamic programming section.

We can observe that the main effect caused by the reduction in the delay is the increase in the number of insertions. This effect limits the usefulness of the phone recognizer to delays of 8 or more frames. Nevertheless, the number of substitutions is not affected by the delay and the number of deletions decreases from a baseline of a 10 % to less than 2 % for a scheme without decoding delay.

These results were not unexpected, the best (delayed) path is more likely to change from one backtracking to the next as the number of frames of delay is reduced. The unrestricted jump from locally optimum paths increases the rate of phones thus reducing phone deletions and increasing phone insertions. On the other hand, the number of substitutions is not affected by the reduction in the delay since the *correct* phone finally appears although its starting frame is delayed by the inserted wrong phones.

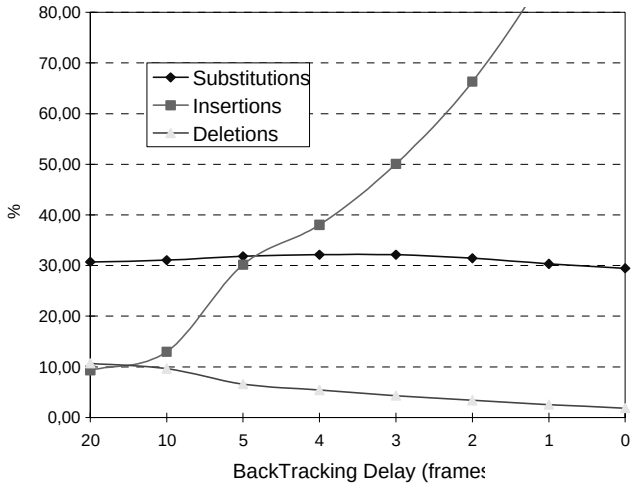

Figure 1: Phone recognition results as a function of the decoding delay. No path pruning.

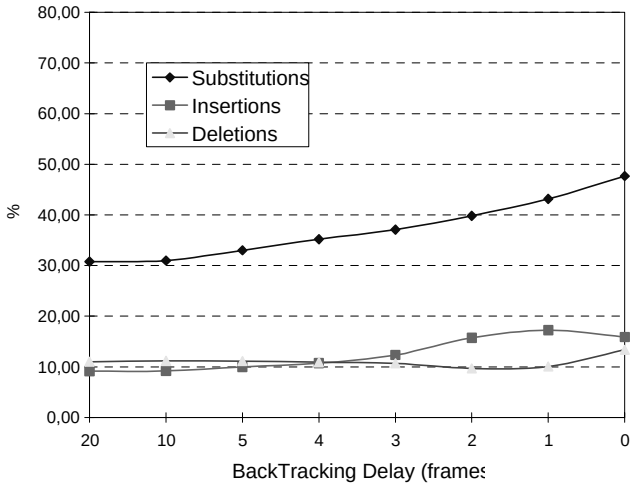

Figure 2: Phone recognition results as a function of the decoding delay. Path pruning.

4.2 Low-delay phone recognition with path pruning

In this second experiment we introduce path pruning to avoid the abrupt increase in the number of phones that appeared in the previous experiment. Figure 2 shows the number of insertions, deletions and substitutions as a function of the decoding delay in frames introduced by the dynamic programming section.

Now, we can observe that the performance degrade smoothly and can be acceptable for most applications even for short decoding delays of only 4 frames.

5 CONCLUSIONS

In this paper, we have discussed the implementation of a low-delay real-time phone recognizer based on the one-step algorithm. The proposed scheme selects at each frame iteration m the state with maximum accumulated log likelihood among all the states of all the phone models. Then, the corresponding path is followed to determine the phone index associated with the frame $m-\Delta m$, where Δm is the desired decoding delay.

For very low delays, this version of the low-delay one-pass algorithm presents erratic jumps between completely different paths causing frequent changes in the estimated phone sequence, (insertions). The paper proposes a *path pruning* algorithm that impose a fixed path fo frames up to index $m - \Delta m$. The resulting algorithm shows then an almost constant rate of insertions and deletions for any delay and it has an acceptable rate of substitutions for decoding delays of 40 ms o less.

References

[1] Carlos M. Ribeiro and Isabel M. Trancoso. "Phonetic Vocoding with Speaker Adaptation". Proc. 5th European Conference on Speech Communication and Technology , pp 1291-1294. Greece 1997.

[2] Mohamed Ismail and Keith Ponting. "Between Recognition and Synthesis - 300 Bits/Second Speech Coding". Proc. 5th European Conference on Speech Communication and Technology , pp 1291-1294. Greece 1997.

[3] VIDAS:
Video Assisted with Audio Coding and Representation. EC ACTS AC057. 1995 - 1997. http://www-dsp.com.dist.unige.it/vidas/index.html.

[4] William Goldenthal, Keith Waters, Jean-Manuel Van Thong, and Oren Glickman. "Driving Synthetic Mouth Gestures: Phonetic Recognition for FaceMe!". Proc. 5th European Conference on Speech Communication and Technology , pp 1995-1998. Greece 1997.

[5] Lawrence Rabiner, Biing-Hwang Juang. *Fundamentals of Speech Recognition*. Prentice Hall 1993.

[6] Chin-Hui Lee, Frank K. Soong K. Paliwal. *Automatic Speech and Speaker Recognition. Advance Topics*. Kluwer Academic Press 1996.

[7] SpeechDat M. Databases for the Creation of Voice Driven Teleservices. EC Telematics LRE-63314. http://www.icp.grenet.fr/SpeechDat/home.html.