Single Speaker Segmentation and Inventory Selection Using Dynamic Time Warping Self Organization and Joint Multigram Mapping

Matthew P. Aylett, Simon King

Centre of Speech Technology Research, University of Edinburgh
Edinburgh, Great Britain
matthewa@inf.ed.ac.uk

Abstract

In speech synthesis the inventory of units is decided by inspection and on the basis of phonological and phonetic expertise. The ephone (or emergent phone) project at CSTR is investigating how self organisation techniques can be applied to build an inventory based on collected acoustic data together with the constraints of a synthesis lexicon. In this paper we will describe a prototype inventory creation method using dynamic time warping (DTW) for acoustic clustering and a joint multigram approach for relating a series of symbols that represent the speech to these emerged units. We initially examined two symbol sets: 1) A baseline of standard phones 2) Orthographic symbols. The success of the approach is evaluated by comparing word boundaries generated by the emergent phones against those created using state-of-the-art HMM segmentation. Initial results suggest the DTW segmentation can match word boundaries with a root mean square error (RMSE) of 35ms. Results from mapping units onto phones resulted in a higher RMSE of 103ms. This error was increased when multiple multigram types were added and when the default unit clustering was altered from 40 (our baseline) to 10. Results for orthographic matching had a higher RMSE of 125ms. To conclude we discuss future work that we believe can reduce this error rate to a level sufficient for the techniques to be applied to a unit selection synthesis system.

Index Terms: speech synthesis, unit selection.

1. Introduction

Recent research in unit selection synthesis has focused on the search problem (finding the optimal unit sequence from the inventory for a target utterance), the prediction problem (how to generate natural sounding pronunciation and prosody for a given utterance in a given context), and the performance/footprint problem (how to compress ever increasing databases and how to speed up ever more complicated join and target cost functions).

However, what we call the unit inventory problem has been neglected. Current systems invariably use conventional phone inventories (although the units may be diphones, half phones, fragments of phones, etc). There remain numerous problems in current systems which we argue are caused by the use of such pre-defined phone sets.

1.1. Problems with manually-specified inventories

The single root cause of the inter-related problems described below is this: describing continuous speech as a linear sequence of phones, drawn from a relatively small and manually-specified inventory, is fraught with problems, Ostendorf's paper "Moving beyond the 'beads-on-a-string' models of speech" is widely cited [1].

Describing continuous speech as a sequence of non-overlapping phones is too simplistic. In reality, phones (the acoustic realisations of phonemes) are not the atomic units of speech - they are subject to variation caused by their context, and this variation is continuous in nature; in other words, when a phone varies away from its canonical form, it does not necessarily change to become the canonical realisation of a different phoneme. More often, certain aspects of the phone change (formants move, voice onset time changes, etc). A description of speech in terms of discrete phoneme categories cannot represent these changes. This is even more of a problem for casual or affective speech where prosodic reduction and prosodic emphasis further increase segmental variation.

Currently, unit selection synthesis uses a set of ad hoc heuristics to deal with problems caused by a manually-specified phone inventory. For example:

Co-articulation Arguably the biggest contribution to phone variance is co-articulation. The typical solution to this problem in speech synthesis is to use diphones to model the speech. One affect of this is to massively increase data-sparsity as we move from a typical inventory of 40 phones to around 1600 diphones. However diphones alone are not sufficient to deal with variance caused by co-articulation. The extent of co-articulation varies and can cross several phone boundaries in extreme cases. Generally a set of ad-hoc rules are added to minimise this problem, for example taking special care not to join a vowel with right 'r' context to ones without such a context.

Vowel and consonant reduction and deletion Reduction occurs naturally and frequently throughout continuous speech. The solution often applied in unit selection is to allow a limited set of discrete pronunciation variants to model reduction and deletion. However the type of reduction and its extent is affected by speaker, speaking style and prosodic structure. Often pronunciation alternatives model this variation quite badly and can lead to errors.

Accent variation For many languages and accents there is no agreed phonetic description. Individual speakers can vary extensively. The variation can be arbitrary, context dependent, and often fundamental for conveying the character and naturalness of the speaker.

Circularity A crucial problem with the unit selection approach is that the phone inventory is used to determine sparsity

and thus the text required for an audio database. Thus developers are required to create phone set inventories before having the audio data from a speaker and before encountering synthesis problems directly dependent on this data. It then becomes resource intensive to re-tune the inventory to optimise the system.

Finally, changes to the inventory have a dramatic impact on the lexicons used for synthesis and the effect of sparsity on the data. Lexicons typically contain many thousands of words and tailoring a lexicon to a specific accent is non-trivial. In turn this makes it hard to alter the phone set. Sparsity is a big problem in unit selection. The amount and type of phones present in the inventory have a dramatic effect on the sparsity. Thus to a large extent the 'ideal' phone set would be dependent on the amount of audio data available in the database. In current system the phone set is fixed no matter how much or how little data is available for a speaker.

1.2. A Machine Learning Paradigm

A separate problem arises from the requirements of so many ad hoc heuristics and so much manual intervention. It becomes impossible to caste the unit selection process into a well defined machine learning problem and thus use constraints and priors in a formalised manner.

In contrast, if the phone inventory can be determined based on a machine learning paradigm it may be easier to extend a machine learning approach throughout the system and make unit selection synthesis much more formalised and more adaptable.

2. Method

In reality, the problem we need to solve is to model the variation for a *single database only* and relate this to a lexicon which can generalise the database to speech that we wish to synthesise. In other words, over fitting a single speaker, a curse in speech recognition, is not a problem for unit selection synthesis. Thus a solution to the inter related voice building problems caused by a manually-specified phone sets can be solved by automatically learning a set of sub-word units. We term this set of sub-word units emergent phones or *ephones* as, unlike a prescriptive phone set, the ephones emerge from the occurrence of regular patterns within the data. By imposing suitable constraints on the properties of these ephones, we can ensure that the resulting set of ephones, and the corresponding ephone inventory, are optimised for use in concatenative speech synthesis.

Figure 1 gives a schematic of how this process could work. First a self organisation method is used for determining a set of ephones, *acoustic ephone selection*. The ephones are then mapped onto a lexicon to produce a *database ephone lexicon*. Phonological rules are then extracted from this database lexicon, and the relationship is generalised to generate ephone transcriptions for all words in the lexicon. The result of this process is then analysed against a set of lexical and acoustic constraints, such as the similarity between generated lexical entries and those aligned in the database, the extent minimal pairs are maintained, the extent sparsity is controlled, and, given a unit selection engine, the extent the system generates acoustic stability for joining units. The results of this analysis are then used as constraints and priors to further improve the initial acoustic ephone selection.

The work we report here is concerned only with the initial acoustic ephone selection and the creation of the initial database lexicon.

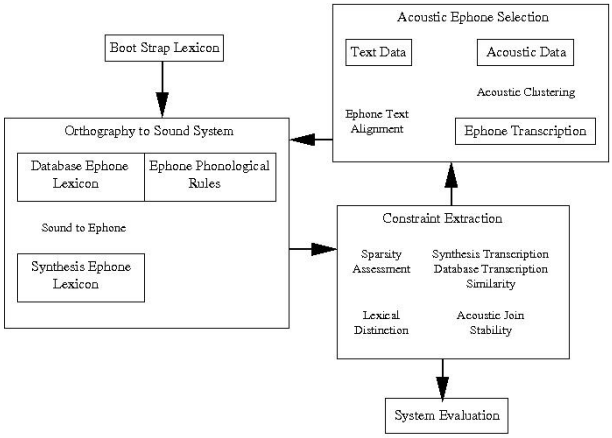

Figure 1: *A three stage machine learning process for unit selection voice building using ephones.*

2.1. Acoustic ephone selection

2.1.1. Segmentation

Automatically determining the phone set used to describe speech already has been examined, with some success, in speech recognition research (e.g [2, 3, 4, 5]), In this paper we focus on an approach using dynamic time warping (DTW) to find repeated patterns in speech and use these as ephones.

This approach is inspired by work by Park and Glass in speech pattern discovery [6]. We may regard a good unit of speech as a pattern that occurs regularly across the speech stream. A method for determining these patterns is to compare each utterances with all other utterances and find patterns that often co-occur.

Figure 2 shows how this comparison is accomplished. A full two dimensional comparison matrix is constructed with each cell containing the result of a distance calculation between every frame of speech in the first utterance and every frame of speech in the second utterance.

In the experiments reported here the speech was parametrised into 10ms frames containing 12 MFCCs and an energy component. All parameters were normalised and then the energy component was increased in size by a factor of ten. In initial studies this was found to improve the classification of silent sections of the speech. A Euclidean distance metric was used.

The algorithm then iterates down one side of the matrix and analyses the diagonal starting at this position. Three parameters are used to determine 'matching sections' within the diagonal:

1. A maximum threshold for the average comparison distance allowed over a matching segment.
2. A minimum time for a matching segment.
3. A maximum distortion allowed over the matching segment, expressed as the width W of the diagonal that the DTW algorithm is permitted to use (see figure 2).

A DTW path is computed along the permitted diagonal. Sections greater than the minimum length and with an average comparison below a threshold are then retained. We chose a a minimum length of 10ms, a maximum distortion of 210ms and

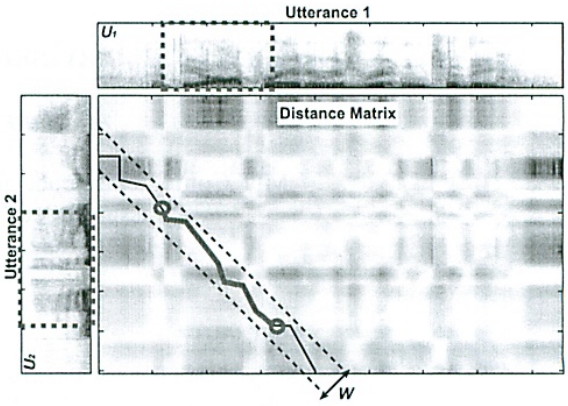

Figure 2: *Using dynamic time warping (DTW) to find co-occurring patterns in two utterances. w is the distortion allowed during the match. The bold line shows a section of the matching path where the average match is below the required threshold. (Taken from [6] p54).*

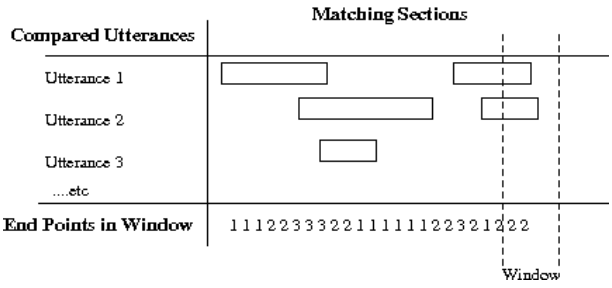

Figure 3: *The number of matching section end points are computed in a window. Maxima of number of end points present in a window are then used to place ephone boundaries.*

a maximum average comparison distance of 3.5. All sections found in this way are then written to a results file.

The results of this file were then analysed for section boundaries. A window of 50ms was passed over all sections. The number of sections starting and ending in the window were summed. This produced a parametric value that was high for frames in the speech, where matching sections terminated and others began. Figure 3 shows a schematic of this scoring process. A peak picking algorithm was then applied to this data by passing a window of 90ms across the result and placing a ephone boundary where the centre of this window was a maxima with regards to its full left and right context.

Figure 4 shows the result of this segmentation on the words 'for real change in' taken from the phrase 'we're also looking for real change in public behaviour' and comparing it to a traditional hidden markov model (HMM) segmentation carried out using HTK[7].

It is worth noting that this process is not the same as using a discontinuity metric. The matching segments may (and do) contain sharp spectral changes. However these are changes which are repeated throughout the data in similar contexts. The boundaries that this process finds are where no consistent matching sections were found. Arguably such a boundary marks a transition between matching regions and thus a location of a ephone boundary.

This segmentation process is processor intensive as it is quadratic with regards to database size. We applied this technique to a single database recorded at CSTR as part of the Festival unit selection system. The speaker was a young RP accented woman and the database we examined consisted of 728 utterances, 13k words, 60k phones, 1.8 hours of total speech and 1.38 hours of total phonetic material (total speech time with silence subtracted). This was approximately a third of the total database but is similar in size to many small unit selection databases.

To reduce computation time a reference set of utterances were selected to compare with all others. These were selected on the basis of entropy. The higher the entropy of the parameter distributions in the utterance, potentially, the more the variation within it. For example an utterance file of complete silence would have a low entropy where as an utterance of babble would have a high entropy. Utterances with the highest entropy scores and a total combined duration of not more than 200 seconds were selected as reference speech.

2.1.2. Ephone identity

In order to group segmented ephones we carried out a k means clustering using ephones as medoids. This was carried out on two numbers of clusters, 40 and 10. Once the reference data was clustered all ephones were grouped according to this initial clustering. The same dynamic time warping metric was used to compare clusters as was used initially in the segmentation.

We envisage this k-means clustering approach to be used as a baseline for further work. In further systems we expect the number of clusters to reflect the variation in the data rather than be set in advance.

Every ephone was then named according to its relationship with the baseline HMM) segmentation. For each ephone the phone that overlapped with the greatest number of frames was chosen as a name for the ephone together with the percentage of this overlap and the overall duration of the ephone in frames.

Clusters were named based on the largest set of member ephones with the same associated HMM based phone name, together with a three digit index.. The largest clusters were named first with an index of '000'. Smaller clusters with the same majority phone content were named with the phone and an incremented index.

Figure 4c shows an example of the words 'for real change in' with the ephones are labelled by cluster name. Care is required when interpreting the names of clusters. For example the first ephone '@0:002' is named as such because the majority of the ephones in the cluster mostly overlapped a '@0' (unstressed schwa) in the HMM segmentation. However this is the third largest cluster of this kind and given it contains unvoiced frication suggests it represents mostly elided schwas with heavy contextual frication.

2.2. Initial database lexicon

The creation of the ephone inventory is completely driven by bottom up processing. In order to carry out synthesis with the ephones we need to relate sequences of ephones to the words we wish to synthesise. These words can be regarded as a string of symbols. Given the vagaries of English spelling it was decided to use two alternative sequences: 1) The lower case letters themselves without hyphens, apostrophes or capitalisation.

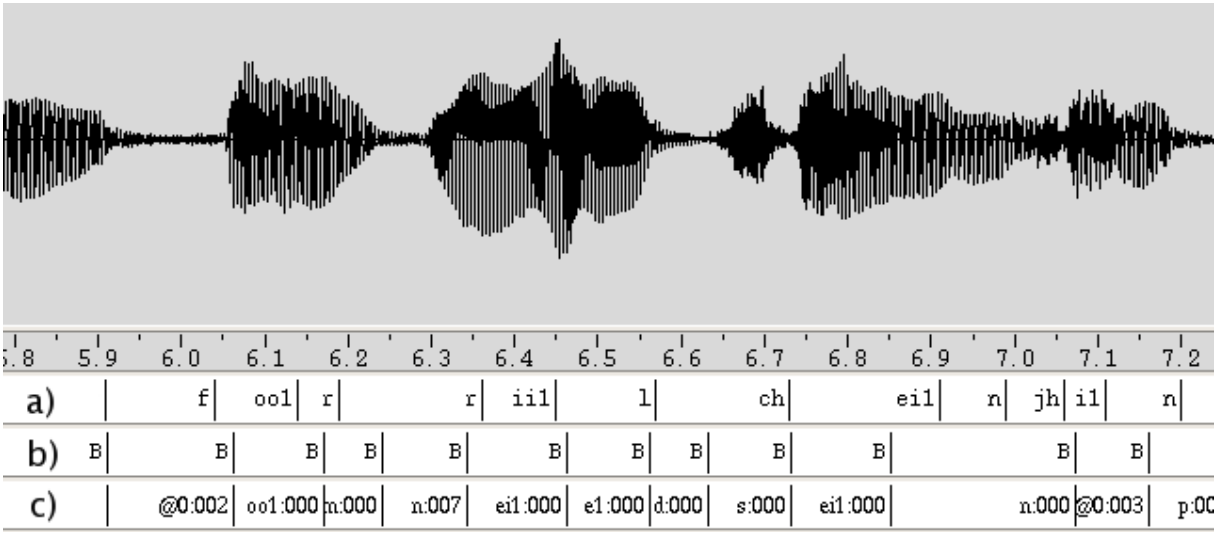

Figure 4: *Example of a) segmentation carried out by HTK b) boundaries proposed by DTW segmentation c) ephone names proposed by clustering. The sequence is "for real change in" taken from the phrase "but we're also looking for real change in public behaviour".*

2) The pronunciation of the word in terms of phone sequences from the traditional HMM segmentation. The phone set acts as a 'best case' baseline in that the acoustics of the words should relate more closely to the phone series than the letter series.

In this study we used silence detection and location information from the traditional HMM segmentation to reduce the degrees of freedom within the system. In the long term, silence insertion will need to be modelled in any sequence matching system.

We applied a joint multigram approach to matching sequences together based on work by [8]. We chose a joint multigram formalism because it allows multiple to multiple matching between letter/phone sequences and the emergent phones.

The multigram model was originally developed by Bimbot et al in order to model variable length regularities within streams of symbols hence the term *multigram* as opposed to *n-grams*. The joint multigram [9] relates two multigrams from separate streams and can be used to segment two streams into concurrent multigrams.

See [8] for a full description. Briefly, the probabilities of each multigram are recalculated based on a set of co-occurring streams using expectation maximisation. Observed probabilities are calculated using the forward backward algorithm. For example Table 1 shows the result of this process when applied to the problem of segmenting letters and phones. The result is to split the phones and the orthography into morphologically appropriate sequences.

The process for matching letter or phone sequences and ephones is made more difficult because the sequences are much longer and thus the number of possible multigram segmentations can be very large. Table 2 shows the result of applying the joint multigram algorithm to the speech in figure 4. The word boundaries are, in most part, the closest ephone boundaries to the phone word boundaries, except where the ephone n:000 at the end of the word 'change' has been co-segmented with the 'i' in the word 'in'. These types of co-segmentation error could have a serious impact on synthesis quality using this segmentation.

Table 1: *Using joint multigrams to co-segment letters and phones in a pronunciation dictionary. (MRPA phone set)*

Word	Pronunciation.	Letter Sequence	Phone Sequence
accompany	@ k uh m p @ n ii	ac com p any	@ k uh m p @ n ii
accomplice	@ k o m p l @ s	ac com pl ice	@ k o m p l @ s
accomplish	@ k o m p l i sh	ac com pl ish	@ k o m' p l i sh
accounts	@ k au n t s	acc oun ts	@ k au n t s

3. Results

Although the traditional HMM segmentation suffers from many of the problems we are expressly trying to address with the techniques described here, it can still act as an effective means of evaluation. Although we would not expect a perfect ephone segmentation to match boundaries in a traditional segmentation we would not expect boundaries to be grossly different in many locations. This is especially true at word boundaries.

If we compare the closest ephone boundary to each word boundary in the HMM segmentation the root mean square error (RMSE) of this comparison is 35ms. Thus 95% of all boundaries in this best case comparision are within 70ms of the traditional HMM word segmentation. Currently, without a perceptual test, we do not know whether the HMM boundary or the ephone boundary is correct and given this uncertainty such a

Table 2: *Using joint multigrams to co-segment a phone sequence and an ephone sequence. See figure 4 to compare word end times to the HTK segmentation.*

Word	Phone	EPhone
for	f	@0:002
	oo	oo1:000
	r	m:000
real	r	n:007
	ii	ei1:000
	l	e1:000
change	ch	d:000
	ei	s:000
	n_jh	ei1:000
in	i	n:000
	n	@ 0:003

word boundary error may be acceptable. However in our final system we will not have an HMM segmentation, instead, as we have described in the previous section, we will need to map our units onto a series of symbols, such as orthography, that represents the speech contents.

A means of evaluating the symbol mapping process is as follows:

- Use the HMM phone symbols as a representation of the speech.
- Map these phone symbols onto the ephones.
- Compare the location of the mapped phones with the HMM segmentation (especially at word boundaries).
- If sequence matching is effective we would hope that the error between the mapped phones and the HMM boundaries would approach an RMSE of 35ms which is the best match we could hope for given the ephone segmentation we have produced.

This process can then be compared with the same mapping algorithm but instead applied to orthographic information. By comparing the errors we can assess mapping algorithms, the differences between orthography and a traditional phone set, and the effects of cluster identity. We report results on the following conditions:

1. Matching orthography against traditional phone sequences.
2. Using ephones constructed with 40 and 10 clusters.
3. Varying the multigrams allowed. For example we can describes a joint multigram as *1-1*, where one symbol only matches one ephone, or *2-1* where two symbols match one ephone and so on. The ratio of phones to ephones and letters to ephones is respectively 1.4 and 1.9. Therefore a mixture of 1-1 and 2-1 multigrams are the minimum types required to allow a match between sequences. We then added further multigram types to see if this increased or decreased word boundary error.

Table 3 shows results for all phone boundaries for the phone matching conditions and for word boundaries for all conditions.

Table 3: *Root mean square error (RMSE) between word boundaries proposed by an ephone segmentation and a baseline HTK segmentation. Multigram types are expressed as [no. symbols]:[no. of ephones]*

clusters: 40, Phones, Multigrams 1-1, 2-1 All Boundaries: *RMSE 104ms* Word Boundaries: *RMSE 0.102ms*
clusters: 10, Phones, Multigrams 1-1, 2-1 All Boundaries: *RMSE 126ms* Word Boundaries: *RMSE 124ms*
clusters: 40, Phones, Multigrams 1-1, 2-1, 1-2, 2-2 All Boundaries: *RMSE 116ms* Word Boundaries: *RMSE 109ms*
clusters: 40, Letters, Multigrams 1-1, 2-1 Word Boundaries: *RMSE 126ms*
clusters: 40, Letters, Multigrams 1-1, 2-1, 1-2, 2-2, 3-1 Word Boundaries: *RMSE 131ms*

4. Discussion

The sequences we are trying to co-segment are quite long compared to word/pronunciation sequences shown in table 1. The average length of each speech chunk separated by silence is 22 ephones (standard deviation = 13). Segmenting the words to within 100 to 200ms would be regarded as quite good for say a search application, especially given no phone model is required. However the results from the joint multigram co-segmentation are significantly worse than the best case of matching closest ephone boundary to closest word boundary. In addition this granularity is too poor for unit selection synthesis where an error of much more than a phone size will cause the addition of unwanted acoustics or the loss of required acoustics.

As expected using a lower cluster size for the ephones resulted in worse performance. However the additional multigram types, for example 1-2, 2-2, 3-1 for orthographic mapping, reduces the performance. We believe there may be two reasons for this:

1. The extra multigram types are over fitting and the data.
2. The lack of a duration penalty. A 2-2 letter to ephone match is not regarded as having an intrinsic cost for crossing 2 boundaries. Thus in most cases longer multigrams are selected over shorter multigrams. This in turn contributes to data sparsity and poor co-segmentation.

However we believe the use of word boundary as an evaluation metric will allow us to improve the co-segmentation, perhaps with the addition of priors relating duration to the multigram identity. If the co-segmentation is improved it then becomes possible to improve the self-organisation and clustering approach to the acoustic segmentation.

This work is still in its early stages. Currently a set of engineering decisions have been made purely to generate a working baseline and a working evaluation of this baseline. Although the segmentation may not be ideal, it is the ephone identity derived from the clustering process and the sequence matching between these derived ephones which requires most improvement. We, believe, with the use of an automatic evaluation criteria that these processes can be improved. In future work we expect to consider ergodic HMMs as a clustering process, using Bayesian information criteria (BIC) to select cluster sizes and number, and looking more deeply into the effect of the parameters used in the current model on the segmentation and inventory selection.

5. Acknowledgements

Support for this research was provided by EPSRC (award number EP/D058139/1).

6. References

[1] M. Ostendorf, "Moving beyond the 'beads-on-a-string' model of speech," in *IEEE ASRU*, 1999.

[2] T. Holter and T. Svendsen, "Incorporation linguistic knowledge and automatic baseform generation in acoustic subword unit based speech recognition," in *Proceedings of Eurospeech 97*, 1997, pp. 1159–62.

[3] M. Bacchiani and M. Ostendorf, "Joint lexicon, acoustic unit inventory and model design," *Speech Communication*, vol. 29, pp. 99–114, 1999.

[4] R. Singh, B. Raj, and R. Stern, "Automatic generation of sub-word units for speech recognition systems," *IEEE Transactions on Speech and Audio Processing*, vol. 10(2), pp. 89–99, 2002.

[5] S. Chen and P.S.Gopalakrishnan, "Clustering via the bayesian information criterion with applications in speech recognition," in *ICASSP 98*, 1998, pp. 645–8.

[6] A. Park and J. Glass, "Towards unsupervised pattern discovery in speech," in *IEEE ASRU*, 2005, pp. 53–58.

[7] S. Young, J. Jansen, J. Odell, D. Ollason, and P. Woodland, *The HTK Book*. Entropic, 1996, version 2.00.

[8] S. Deligne and F. Bimbot, "Inference of variable-length linguistic and acoustic units by multigrams," *Speech Communication*, vol. 23, pp. 223–241, 1997.

[9] S. Deligne, F. Yvon, and F. Bimbot, "Variable-length sequence matching for phonetic transcription using joint multigrams," in *Eurospeech*, vol. 3, 1995, pp. 169–172.