



# Audio analytics by template modeling and 1-pass DP based decoding

Srikanth Cherla, V. Ramasubramanian

Siemens Corporate Research & Technologies - India, Bangalore, India

Srikanth.Cherla@siemens.com, V.Ramasubramanian@siemens.com

## Abstract

We propose a novel technique for audio analytics and audio indexing using template based modeling of audio classes set in a one-pass dynamic programming continuous decoding framework. We propose use of concatenation costs in the one-pass DP recursions to reduce so-called incursion errors; we also propose selection of variable length templates for modeling indefinite duration audio classes using the segmental  $K$ -means (SKM) algorithm. Based on detailed decoding results with long audio streams, we conclude the effectiveness of template based modeling, SKM based template selection, 1-pass DP based decoding and the use of concatenation constraints therein. We show that an average (%Hit, %False-alarm) of (66%, 4.9%) are possible with the proposed decoding technique.

**Index Terms:** Audio analytics, audio indexing, audio classification, audio segmentation, template modeling, continuous decoding

## 1. Introduction

In this paper, we address the problem of audio analytics in the context of surveillance application where it is required to recognize various audio events (classes) in an audio stream, typically from a camera deployed in a place of interest (street, retail, station, airport etc.). Audio classification has so far been done using various types of modeling and classification, such as  $K$ -nearest-neighbor ( $K$ -nn) classification [1], [2] decision tree classification [2], quadratic Gaussian classifier [2], GMM [1] and HMM [3], [4], [5], [7], [6].

In this paper, we propose an alternate audio class modeling by means of templates of the audio class. This belongs to the non-parametric modeling approach, which has received some attention recently in the context of speech recognition [9], [10]. Within this framework of template based audio class modeling, we formulate the problem of audio analytics as a ‘connected audio recognition’ problem (in line with the erstwhile connected word recognition problem in speech recognition), and propose a modified form of one-pass dynamic programming (DP) [8] for decoding the input audio stream to derive an optimal segmentation and labeling as required in a typical indexing system. This marks a departure from earlier work which has either studied only the problem of classifying audio sounds in the form of ‘isolated instances’ [2, 1, 7, 3, 4, 5] or handled the problem of audio segmentation and classification using heuristic rules with thresholds on audio features and which are restricted to only a small number of broad audio categories [11], [12]. In contrast, our formulation of audio ‘decoding’ by 1-pass DP here provides a rigor and consistency which is difficult to achieve with rules and thresholds; moreover, we extend the decoding to a large number of audio classes (up to 15) as is typically of interest in a practical surveillance setting.

## 2. Template modeling

In proposing to model audio classes using multiple templates, we make the observation that there are two distinct types of classes depending on the nature of the audio event: i) **Definite duration classes:** These are audio events characterized by a definite start and end and hence a specific duration within which the event occurs; examples are glass-breaking, vehicle-passby, single vehicle-horn, vehicle-skid, vehicle crash etc., and ii) **Indefinite duration classes:** These classes are not constrained to have a definite duration and can typically extend over any length of time; these are typically continuous in nature and can occur for a duration limited only by its occurrence in a specific instance; examples are babble, footsteps, market etc. These are not usually divisible further into smaller constituent audio elements and are best characterized by a continuous evolution of non-stationary acoustic spectra, though with an overall spectral signature that is self-similar over any length of occurrence.

We model these two categories of audio classes in different ways: i) **Whole templates:** The definite duration audio class is best modeled as whole templates, where a whole template is a single occurrence of the event, such as glass-breaking. Multiple whole templates of such an audio class effectively model the intra class variability (including non-linear temporal variability) even while preserving the acoustic signature of the class as it occurs over the definite duration inherent to the class, and ii) **Bag of templates:** The indefinite duration audio classes are modeled by a bag of short templates. The short templates could be fixed or variable length segments cut out from the long continuous audio class (ex: babble, chopped into 10 frame long segments); an appropriate number of these (say, 50 per class) adequately models an audio class by virtue of sampling the acoustic space spanned by this class, even while retaining the temporal signature of the audio class in the form of quasi-stationary short templates.

## 3. One-pass DP decoding

Given an input audio stream, consisting of a sequence of audio classes, audio analytics or indexing requires a segmentation and labeling of the audio stream accurately in terms of the audio classes comprising the pre-specified audio-vocabulary. Fig. 1 shows a typical solution arising out of posing this problem as a ‘connected audio recognition’ problem. The  $x$ -axis is the input audio (CarPassby-Market-CarSkid) and the  $y$ -axis represents the audio-vocabulary with 4 classes shown here (CarSkid, Market, CarPassby and CarCrash), each modeled appropriately by whole-templates (CarSkid, CarPassby, CarCrash) or a bag-of-templates (Market). The figure shows a typical solution to this problem when treated as a continuous decoding of the input audio into the constituent audio classes. The warping path shown is the optimal solution to this decoding problem and implicitly solves for the segment boundaries and also labels

them. Specifically, note the way the ‘Market’ segment in the input gets decoded by a series of templates drawn from the bag-of-templates of ‘Market’. The whole template of CarSkid or CarPassby warps over its full duration to match with the input segments of these classes. We now consider the formulation of this decoding solution in the one-pass dynamic programming (DP) framework.

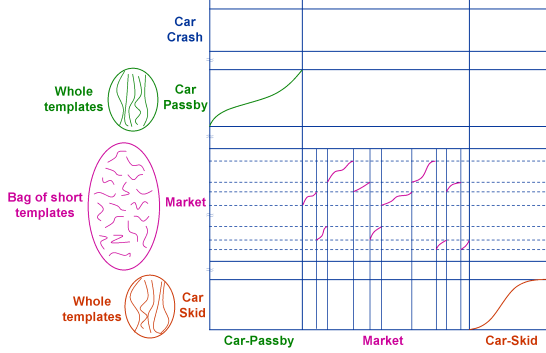


Figure 1: Continuous audio decoding with multiple template modeling: whole templates and bag-of-templates

Let the audio class vocabulary be  $\mathcal{A} = (a_1, a_2, \dots, a_i, \dots, a_V)$ . Each audio class  $a_i$  is represented by  $M_i$  multiple templates (whole or bag) given by  $(a_{i1}, a_{i2}, \dots, a_{im}, \dots, a_{iM_i})$ . Each of these templates  $a_{im}, m = 1, \dots, M_i$  has  $N_{im}$  frames, i.e., is a sequence of  $N_{im}$  MFCC parameter vectors  $(\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_n, \dots, \mathbf{o}_{N_{im}})$ .

Let the input audio which is to be decoded (segmented and labeled) using the above multiple-template models be a sequence of vectors (say, MFCC parameters)  $\mathbf{O} = (\mathbf{o}_1, \mathbf{o}_2, \dots, \mathbf{o}_T)$ . Such a continuous decoding, in its most general form, involves segmenting and labeling this sequence of vectors  $\mathbf{O}$  by a ‘decoding’ or ‘connected segment recognition’ algorithm which optimally segments the sequence and labels each segment by an appropriate label or index from the template models.

Consider an arbitrary segmentation of  $\mathbf{O}$  into  $K$  segments  $S = (s_1, s_2, \dots, s_{k-1}, s_k, \dots, s_K)$ . This segmentation can be specified by the segment boundaries  $B = ((b_0 = 0), b_1, b_2, \dots, b_{k-1}, b_k, \dots, (b_K = T))$ , such that the  $k^{th}$  segment  $s_k$  is given by  $s_k = (\mathbf{o}_{b_{k-1}+1}, \dots, \mathbf{o}_{b_k})$ . Let each segment be associated with a label from the audio vocabulary; let this index sequence be  $Q = q_1, q_2, \dots, q_{k-1}, q_k, \dots, q_K$ . Each index  $q_k \in \{1, \dots, V\}$ .

An optimal decoding involves solving for the optimal solutions of  $(K, B, Q)$ , i.e.,  $K^*, B^*, Q^*$  corresponding to the minimum overall distortion  $D^*$  given by

$$D^* = \min_{K, B, Q} \sum_{k=1}^K D(s_k, a_{q_k}) \quad (1)$$

Here,  $D(s_k, a_{q_k})$  is the distortion in labeling segment  $s_k$  as audio class  $a_{q_k}$ . For the case of multiple template modeling of each audio class, this distortion is essentially a dynamic time warping distance between  $s_k$  and the template of  $a_{q_k}$  which has the minimum distortion with  $s_k$ .

The optimal  $(K^*, B^*, Q^*)$  are solved jointly by minimizing  $D^*$  over all possible  $(K, B, Q)$  which is solved efficiently by using one-pass dynamic programming (DP). We now describe the proposed one-pass DP algorithm for continuous audio decoding with multiple template based acoustic modeling of audio classes. As described earlier, the multiple templates can either be whole templates or bag of templates; the one-pass

DP based decoding described here handles both these categories of template modeling and yields an appropriate decoding. The recursions used here are a modified form with respect to the conventional connected word recognition [8], to account for the multiple template based modeling of each audio class.

In the different recursions shown here, the one-pass DP calculates the optimal (minimum) accumulated distortion  $D(t, n, m, i)$  to reach the  $n^{th}$  frame of template  $m$  of audio class  $i$ , at every time instant  $t = 1, \dots, T$  of the input continuous audio data. The local distance  $d(t, n, m, i)$  in these recursions is the distance between the  $t^{th}$  frame of the input audio and the  $n^{th}$  frame of template  $m$  of audio class  $i$ . The recursions of the proposed one-pass DP algorithm for continuous audio decoding using multiple templates are as follows.

### 1. Within-class recursion

This recursion is applied for each of the multiple templates of each audio class; these are applied for all frames that are not template-beginning frames, i.e., calculate  $D(t, n, m, i)$  only for the template-interior frames  $n = 2, \dots, N_{im}$ , for all templates  $m = 1, \dots, M_i$  of an audio class  $i$ , for all audio classes  $i = 1, \dots, V$  and for every time instant  $t = 1, \dots, T$ .

$$D(t, n, m, i) = d(t, n, m, i) + \min_{j=(n, n-1, n-2) \& (j>0)} [D(t-1, j, m, i)] \quad (2)$$

### 2. Cross-class recursion

This transition recursion corresponds to entry into the first frame  $n = 1$  of any of the  $M_i$  multiple templates  $m = 1, \dots, M_i$  of audio class  $i$  from (the last frame  $N_{jm}$  of) any of the  $M_j$  multiple templates  $m = 1, \dots, M_j$  of all audio classes  $j = 1, \dots, V$ , i.e., calculate  $D(t, n = 1, m, i)$  for every time instant  $t = 1, \dots, T$ , for  $n = 1, m = 1, \dots, M_i$  and  $i = 1, \dots, V$  as,

$$D(t, n = 1, m, i) = d(t, n = 1, m, i) + \min [D(t-1, n = 1, m, i), \min_{j=1, \dots, V} [\min_{m=1, \dots, M_j} D(t-1, N_{jm}, m, j) + \Delta_{ji}]] \quad (3)$$

$\Delta_{ji}$  is 0 for the baseline algorithm and has a specific interpretation and role as a concatenation cost to handle a type of error referred to as ‘incursions’; this is described further in Sec 3.1.

### 3. Termination and backtracking

The decoding using the above recursions terminates at the last time instant  $T$  with the optimal accumulated distortion  $D^*$ ,

$$D^* = \min_{i=1, \dots, V} \min_{m=1, \dots, M_i} D(T, N_{im}, m, i) \quad (4)$$

i.e., this is the minimum accumulated distortion over the last frames  $N_{im}$  of all the  $M_i$  templates of all the audio classes  $i = 1, \dots, V$ . The optimal path through the one-pass DP ‘grid’ and the corresponding decoded audio class sequence are recovered by backtracking using the backpointers stored during the forward computations of the recursions given by Eqns. (2) and (3). The backpointer equations and the backtracking equations are not shown here due to space constraints.

#### 3.1. Incursion errors and concatenation costs

We first show results for the one-pass DP based decoding using a sample audio data with the audio classes ‘market-carpassby-market-carpassby-market’ occurring in a sequence of total duration 25 secs. This is shown in Fig. 2. Here the decoding is done with the following vocabulary of audio classes: whole template classes - ambulance, bicycle bell, car passby, car rev,

car crash, horn, police car, truck passby and bag-of-template classes - footsteps, children playing, market, traffic jam and skid. The top color strip in Fig. 2(a) (and (b)) is the ground truth and the adjoining bottom color strip in Fig. 2(a) (and (b)) is the decoded result.

The result shows a type of error which we refer to as ‘incursions’. These are mainly a series of short misclassifications of an indefinite duration class (market, in this case) as some other class (ex: footsteps, skid). This type of error occurs mainly due to the provision of cross-class recursions to allow for a transition from one class to another. However, this provision (a necessary one for valid cross-class transitions) results in a series of erroneous short transitions from a wrong class into a given class. Ideally, it can be seen that if the input audio contains an indefinite duration class such as ‘market’, then the correct decoding of this should be as illustrated in Fig. 1 where the multiple templates from the bag-of-templates from ‘market’ are chosen in some order successively to decode the input segment corresponding to ‘market’. This requires multiple series of transitions from ‘market’ to ‘market’ i.e., across class transition, but happening as self-transitions of the same class.

Based on this observation, a solution to the ‘incursions’ is to facilitate or favor such self-transitions, i.e., whenever a cross-class transition has to happen from class ‘i’ to ‘i’. For this, the  $\Delta_{ji}$  in Eqn. (3) is included as a ‘concatenation cost’ and is defined as follows:

$$\begin{aligned} \Delta_{ji} &= 0 \text{ if } j = i \\ &= d(\mathbf{o}_{N_{jm}}^j, \mathbf{o}_1^i) \text{ if } j \neq i \end{aligned}$$

Fig. 2 shows the results for the audio sequence ‘market-carpassby-market-carpassby-market’ including the above concatenation cost. It can be seen that using the concatenation cost significantly reduces the ‘incursion’ errors from other bag-of-template classes, i.e., the large number of incursions in ‘Market’ (without concatenation) are more or less completely removed with the use of concatenation. This is due to the following reasons: A 0 cost when  $j = i$  favors self-class transitions such as is required to correctly decode ‘market’ and as shown in its ideal form in Fig. 1. A cost of  $d$  (when  $j \neq i$ ) is the acoustic cost between the last frame of one class ( $j$ ) to the first frame of the current class ( $i$ ) and weighs such non-self cross-class transitions more and favors them less, thereby reducing ‘incursion’ type of errors such as between any wrong bag-of-template class (ex: footsteps) to the correct bag-of-template class (‘market’ in the example shown).

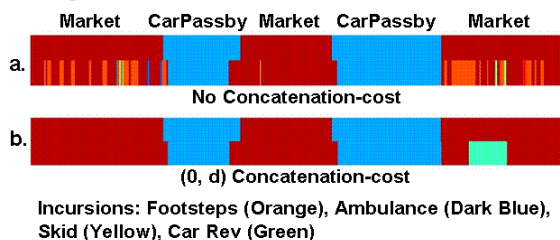


Figure 2: Decoding performance with concatenation costs

#### 4. Segmental K-means template training

Our first experiments with bag-of-template modeling is to select a fixed number (say, 50 or 100) of short templates (all of same fixed length, say 10 frame long or 100 ms duration). The motivation for this is that an arbitrary test audio can be approximated by such short templates in a piecewise manner, even while retaining some temporal signature of the overall class in its short

templates, thereby resulting in good match and decoding. On the other hand, very short templates of one class can easily make an ‘incursion’ error into another class, since short templates may not retain the unique signature of a class unambiguously. This therefore points to possible use of longer templates to make up the bag of templates; ideally, these templates can be variable length and perhaps selected by means of clustering on a large amount of training data representative of the audio class, so that the resulting templates sample the acoustic space spanned by the audio class in such a manner as to decode arbitrary test audio of this class effectively and cause no incursion errors into test audio of other classes.

Such a bag-of-template approach to modeling indefinite duration audio classes is essentially similar to the use of variable-length segment codebooks in segment vocoders for segment quantization of speech [13]. Here, we use the variable-length segment quantization training algorithm referred to as ‘joint segmentation and quantization’ [13] or more commonly later as ‘segmental K-means’ (SKM) algorithm [14] for the design of variable length bag-of-templates (also referred to as ‘codebook’ here) of sizes  $P = 5, 10, 20, 30, 40$  and 50.

Such a SKM algorithm yields a codebook (bag-of-templates) of the desired size  $P$  where the templates are variable-length. We incur several advantages due to this SKM based template selection over random templates: i) since the SKM is a clustering algorithm, a SKM derived  $P$  templates are more optimal than randomly selected  $P$  templates in modeling the distribution of any audio class; this is borne out by the distortion convergence over iteration, where the initial codebook (randomly selected fixed length codebook) has high distortion which reduces with iteration leading to the final variable-length codebook with much lower average distortion (in decoding the training data by the codebook); this in turn is equivalent to better decoding performance as is already evident by the average distortion, ii) viewed as a non-parametric modeling of the audio class distribution, the SKM templates decode unseen test audio better due to better generalizability, iii) use of variable length (and hence longer, and more optimal ones) templates have less incursions into another class, since the acoustic signature of a class is better captured and rendered less ambiguous.

#### 5. Experiments and Results

We derived SKM templates for 8 bag-of-templates audio classes, namely, Angry crowd (52.6), Babble (119.9), Bar babble (153), Station Babble (147.7), Children (155.3), Traffic Jam (64.8), Market (154.1), Public Announcement (34.5); the numbers in parenthesis are the amount of training data used for SKM training in seconds. The training and test data for these experiments were obtained from various audio databases: BBC Sound Effects Library, Series 1000 and Series 6000 [15]. In Fig. 3, we show the results of the decoding (with respect to ground truth) for one long audio data (15 sec long, comprising of 15 audio class events of 1 sec duration each) for 4 cases: Random templates a) without concatenation costs and b) with concatenation cost; SKM templates c) without concatenation costs and d) with concatenation cost. In each sub-plot (ex: Random Templates (5)), the top color strip is the ground truth (one color for each of the 8 classes making up the 15 events) and the bottom color strip is the decoded result.

The following can be noted from this figure: i) The random template performance for case (a) is quite poor, with high incursion errors for both smaller (5-20) and larger (50, 100) codebook sizes; ii) Using concatenation costs (case (b)) helps the random template performance significantly, though there

continue to be some incursion errors; iii) SKM templates even without concatenation costs (case (c)) reduce the incursion errors significantly, when compared to random templates (case (a)) and shows the effectiveness of SKM selection for any given codebook size, iv) Using concatenation costs with SKM templates (case (d)) further enhances the performance by removing the marginal incursions in case (c), and offers the best performance among all 4 cases. Case (d) is also naturally better than Case (b) showing the effectiveness of SKM selection of variable-length templates over random selection.

In order to quantify the decoding results further and provide a comparison of the above 4 cases, we have decoded 5 long audio streams each 15 secs long and with 15 1sec individual audio-classes and obtained the ROC measures (%Hit, %False-alarm) as an average over the 5 long audio test data. Table 1 shows the average (%Hit, %False-alarm) for the 4 different cases considered here: Different codebook sizes: {50, 100} for random templates and {5, 10, 20, 30, 40, 50} for SKM derived templates and for without and with concatenation costs.

The following can be noted from this table: i) the observations made above with respect to Fig. 3 continue to hold here and brings out the effectiveness of concatenation cost in improving the performance of random templates, ii) SKM derived templates do well even without concatenation costs and SKM 50 has better ROC measure than Random 50; with concatenation costs, SKM does not derive as much performance gain as the random templates, iii) SKM derived templates show an overall trend of improvement in both (%Hit, %False-alarm) with increase in codebook size, iv) SKM 50 without concatenation cost gives a comparable performance to Random 50 or 100 (with concatenation costs) showing that the clustered training for variable template selection is effective in not requiring concatenation costs due to their inherent long templates (the 10 frame short random templates incur high incursion errors without concatenation costs, which in turn reflects in poorer %Hit and %False-alarm), v) the proposed decoding using templates (be it random or SKM derived) offers an average (%Hit, %False-alarm) of the order of (66%, 5.0%).

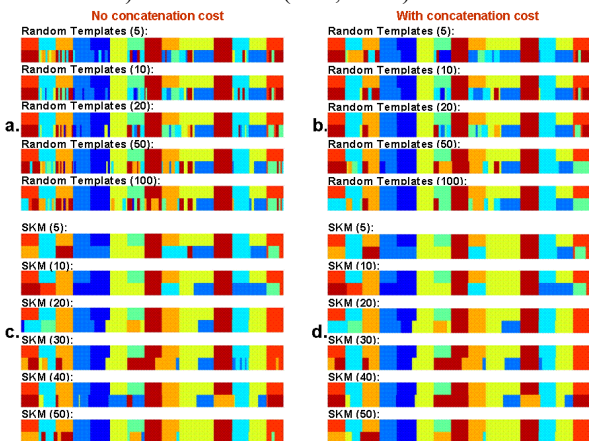


Figure 3: Continuous decoding using Random templates and SKM derived templates without and with concatenation costs.

## 6. Conclusions

We have proposed a novel technique for audio analytics and audio indexing using template based modeling of audio classes set in an one-pass dynamic programming continuous decoding framework. We have investigated concatenation costs to reduce the so-called incursion errors and further proposed template design using segmental K-means (SKM) procedure for se-

Table 1: ROC measures (%Hit, %False-Alarm (%FA)) for template based decoding for random and SKM templates, without and with concatenation costs (Conc-cost)

Trial	No Conc-cost		With Conc-cost	
	%Hit	%FA	%Hit	%FA
Random 50	62.48	5.58	66.59	5.09
Random 100	57.87	6.15	67.06	5.02
SKM 5	50.98	7.20	51.76	7.03
SKM 10	40.31	8.67	40.65	8.61
SKM 20	60.10	5.78	59.76	5.84
SKM 30	50.73	7.01	51.60	6.94
SKM 40	49.86	7.18	51.13	7.01
SKM 50	66.08	4.90	64.21	5.17

lecting variable length templates for modeling indefinite duration audio-classes. Based on detailed decoding results with long audio streams, we conclude the effectiveness of template based modeling, SKM based template selection, 1-pass DP based decoding and the use of concatenation constraints therein. The proposed techniques offer an average (%Hit, %False-alarm) of (66%, 4.9%).

## 7. References

- [1] V. Peltonen et al. Computational auditory scene recognition. *Proc. ICASSP '02*, Orlando, Florida, 2002.
- [2] K. El-Maleh, A. Samouelian and P. Kabal. Frame-level noise classification in mobile environments. *Proc. ICASSP '99*, 1999.
- [3] P. Gaunard et al. Automatic classification of environmental noise events by hidden Markov models. *Proc. ICASSP '98*, 1998.
- [4] L. Ma, D. J. Smith and B. P. Milner. Context awareness using environmental noise classification. *Proc. Eurospeech '03*, pp. 2237-2240, Geneva, Switzerland, 2003.
- [5] L. Ma, B. Milner and D. Smith. Acoustic environment classification. *ACM Transactions on Speech and Language Processing*, vol. 3, no. 2, pp. 1-22, July 2006.
- [6] P. Nordqvist and A. Leijon. An efficient robust sound classification algorithm for hearing aids. *Journal of Acoustical Society of America*, vol. 115, no. 6, pp. 1-9, June 2004.
- [7] Z. Liu, J. Huang and Y. Wang. Classification of TV programs based on audio information using hidden Markov model. *IEEE Signal Processing Society Workshop on Multimedia Signal Processing*, pp. 27-32, Redondo Beach, CA, Dec 1998.
- [8] H. Ney. The use of one-stage dynamic programming algorithm for connected word recognition. *IEEE Trans. on Acoust., Speech and Signal Proc.*, 32(2):263-271, Apr 1984
- [9] M. De Wachter, M. Matto, K. Demuynck, P. Wambacq, R. Cools and D. Van Compernelle. Template-based continuous speech recognition. In *IEEE Transactions on Audio, Speech and Language Processing*, pp. 1377-1390, vol. 15, no. 4, May 2007.
- [10] V. Ramasubramanian, Kaustubh Kulkarni and Bernhard Kaemmerer. Acoustic modeling by phoneme templates and one-pass DP decoding for continuous speech recognition. *Proc. of ICASSP '08*, pp. 4105-4108, Las Vegas, Mar 2008.
- [11] T. Zhang and C. C. Jay-Kuo. Audio content analysis for online audiovisual data segmentation and classification. *IEEE Trans. on Speech and Audio Proc.*, vol. 9, no. 4, pp. 441-457, May 2001.
- [12] L. Lu, H.-J. Zhang and H. Jiang. Content analysis for audio classification and segmentation. *IEEE Trans. on Speech and Audio Processing*, vol. 10, no. 7, pp. 504-516, Oct 2002.
- [13] Y. Shiraki and M. Honda. LPC speech coding based on variable-length segment quantization. *IEEE Trans. on Acoust., Speech and Signal Proc.*, 36(9):1437-1444, Sept. 1988.
- [14] E. Vidal and A. Marzal. A review and new approaches for automatic segmentation of speech signals. In L. Torres, E. Masgra and M. A. Lagunas (eds), *Signal Processing V: Theories and Applications*, pages 4353. Elsevier Science Publishers B.V., 1990.
- [15] <http://www.sound-ideas.com/bbc.html>, <http://www.sound-ideas.com/1000.html>, <http://www.sound-ideas.com/6000.html>