Morphological Segmentation

Pierre Machart
Master Thesis
ATIAM 2008 - 2009
Université Pierre et Marie Curie, Paris VI

Laboratory : IRCAM - Interactions Musicales Temps Réel
Supervisor : Norbert Schnell

March 2009 - June 2009

Acknowledgments

In the first place, I would like to express my deepest appreciation to Norbert Schnell who accepted me to lead this work in the Real-Time Musical Interaction team and who tutored this work with dedication and enthusiasm.

For their constant support, the time they spent answering my many questions, and their good humor, I thank Julien Bloit and Nicolas Rasamimanana, who fostered my motivation day after day.

The collaboration with composers in this study was especially rewarding. Thus, I would like to thank Roque Rivas for the insight and material he gave me. More importantly, I thank Marco Antonio Suàrez Cifuentes for the fruitful discussion we had, for the patience he had to provide the many manual annotations that were used, and for his constant interest in this project.

In addition, I would like to thank Thierry Artières, Frédéric Bevilacqua, Tommaso Bianco, Arshia Cont and Diemo Schwarz, for their advice and different points of view.

I also thank Baptiste Bohelay, Javier Contreras, Arnaud Dessein, Philippe Esling, Benjamin Levy, John Mandereau and Julien Moumné for the rejoicing lunch breaks we had during 4 months.

Thanks to Emily Bloom-Carlin for spending so much time rereading this thesis with constant patience and support.

Abstract

Many applications and practices of working with recorded sounds are based on the segmentation and concatenation of fragments of audio streams. In collaborations with composers and sound artists we have observed that a recurrent musical event or sonic shape is often identified by the temporal evolution of the sound features. We would like to contribute to the development of a novel segmentation method based on the evolution of audio features that can be adapted to a given audio material in interaction with the user.

In the first place, a prototype of a semi-supervised and interactive segmentation tool was implemented. With this prototype, the user provides a partial annotation of the stream he wants to segment. In an interactive loop, the system is able to build models of the morphological classes the user defines. These models will then be used to provide an exhaustive segmentation of the stream, generalizing the annotation of the user.

This achievement relies on the use of Segmental Models, that have been adapted and implemented for sound streams represented by a set of audio descriptors (MFCC). The very novelty of this study is to use real data to build models of the morphological classes, issued from various audio materials. A singular method to build our global model is defined, using both learning paradigms and the integration of user knowledge.

The global approach of this work is validated through experimentations with both synthesized streams and real-world materials (environmental sounds and music pieces). A qualitative and less formal validation also emerges from the feedback given by composers that worked with us along the whole internship.

Résumé

De nombreuses applications et travaux utilisant des sons enregistrés reposent sur la segmentation et la concaténation de fragments de flux audio. A l'occasion de collaborations avec des compositeurs et des artistes, nous avons pu constater qu'un événement musical récurrent ou qu'une forme sonore est souvent identifiable par l'évolution temporelle d'observations du signal. Nous souhaiterions contribuer au développement d'une méthode de segmentation innovante reposant sur l'évolution temporelle de descripteurs audio, et qui s'adapterait aux matériaux audio considérés, en interaction avec l'utilisateur.

En premier lieu, nous avons implémenté un prototype d'outil de segmentation semi-supervisée et interactive. Avec ce prototype, l'utilisateur fournit une annotation partielle du flux qu'il veut segmenter. Dans une boucle d'interaction, le système est alors capable de construire des modèles de classes morphologiques que l'utilisateur définit. Ces modèles sont ensuite utilisés pour proposer une segmentation exhaustive du flux, en généralisant les annotations de l'utilisateur.

Ces résultats reposent sur l'utilisation de Modèles Segmentaux, adaptés et implémentés pour des flux sonores représentés par un ensemble de descripteurs (MFCC). L'originalité de cette étude tient à l'utilisation de données réelles pour construire les classes morphologiques, issues de matériaux sonores divers et variés. Une méthode singulière pour construire le modèle global est alors définie en utilisant à la fois les paradigmes d'apprentissage et l'intégration de connaissances de l'utilisateur.

L'approche globale de ce projet est validée par des expériences menées avec des flux de synthèse ou des sons réels (environnementaux ou des pièces musicales). Une validation plus qualitative et moins formelle tient aussi aux retours donnés par des compositeurs ayant travaillé avec nous tout au long de ce stage.

Contents

List of Figures

Part I

Background and Overview

Chapter 1

Introduction

1.1 Context and Motivation

Many applications and practices of working with recorded sounds are based on the segmentation and concatenation of fragments of audio streams. In particular, recent concatenative synthesis techniques have shown promising possibilities for the synthesis of speech and music [Bloch et al., 2008] [Lindeman, 2007] [Schwarz, 2007] [Collins, 2002] through the concatenation of fragments cut from pre-recorded and analyzed audio streams.

In speech synthesis, these techniques usually rely on large databases of automatically segmented audio material in order to achieve realistic and rich synthesized sounds. The quality of concatenative synthesis depends on the quality of the segmentation as well as the description and classification of the segmented sound fragments.

While the automatic segmentation of speech recordings usually relies on alignment with a text as well as a clear hierarchical structure of phrases, words, syllables and phonemes, the precise and meaningful segmentation and structuring of other sound recordings remains a major challenge.

For recorded performances of an existing musical score and improvised music with clearly identifiable pitches and rhythm, adequate techniques have been developed to align the audio stream to a symbolic music representation [Orio et al., 2003], or to automatically transcribe the melodic and rhythmical structure of the performance [Gouyon and Herrera, 2003] [Klapuri, 1997]. Further techniques can segment arbitrary audio material based on the detection of singularities (i.e. onsets [Duxbury et al., 2003] and silences [Lu et al., 2002]) and recurrences [Peeters, 2004] [Jehan, 2005].

Nevertheless the segmentation of arbitrary complex audio materials us-

ing available techniques often remains unsatisfying, even in cases where the human listener clearly distinguishes a sequence of sound objects of recurrent characteristics and classes. In most professional applications such as recording of acoustic instruments, sampling and sound design, segmentation and concatenation are performed manually to assure the quality of the segmentation as well as the concatenated sound stream.

In collaborations with composers and sound artists we have observed that a recurrent musical event or sonic shape is often identified by the temporal evolution of the sound features within a given segment and that the strategies and criteria to identify a recurrent sound object differ from one sound material to another.

We would like to contribute to the development of a novel segmentation method based on the evolution of audio features that can be adapted to a given audio material in interaction with the user taking into account the following aspects:

- representation and model of the temporal evolutions of sound streams.
- segmentation of sound streams and classification of the sound segments, based on the temporal evolution of sound characteristics.
- exploration of a semi-automatic interactive methods involving the user into the modeling process and taking into account his preferences and knowledge.

We have implemented a prototype of a semi-automatic interactive segmentation system based on Segmental Models (SMs). A user provides a partially segmented and annotated audio stream. From these data, the system will automatically generate an exhaustive segmentation of the stream, generalizing the partial work of the user. The proposed segmentation will eventually be modified by the user. This process ends when the user is satisfied with the provided segmentation.

1.2 Overview

To fulfill these objectives, we wil start with an investigation of the existing work that address the issue of defining musical objects with regards to their temporal evolutions with both musicological and computational point of views in chapter 2. We will then point out some modeling tools that can meet the requirements of this study in chapter 3.

In a second part, we will describe more intensively the specificity of our approach and the contributions that were made. The modeling choices

and implementation will be explained in chapter 4. Then, the interaction paradigm that we explored will be introduced in chapter 5.

Finally, some segmentation results will be given. They will lead to an evaluation of the system and to a discussion of the different assumptions that we made.

Chapter 2

Sound and Music Streams

2.1 Pierre Schaeffer's Musical Objects

In the investigation of the relation between sound, music and semantics, the *Traité des Objets Musicaux* [Schaeffer, 1966], offers a valuable contribution and some rewarding directions of reflection. However, a lot of issues Schaeffer addressed more than forty years ago are still open to debate. An important preliminary point emerges from his early reflections when trying to define the concept of musical object. A musical object is neither its source, nor its musical symbol(s), nor its physical signal. He clearly states that there is no obvious relationship between the signal (or observations from the signal) and the object as it is perceived by the listener.

A more practical synthesis of the ideas that Schaeffer developed is given by Michel Chion [Chion, 1983]. Among them, the ones dealing with morphological criteria will have shed a light on our study. About *matter*, three main criteria have been kept:

- spectral mass: a generalization of the concept of pitch. That roughly describes the distribution of the audible frequencies in the signal.
- harmonic timbre: Schaeffer means *"the additional qualities that seem associated with the spectral mass and allow us to characterize it."*.
- grain : the microstructures of the sound matter.

Concerning the shape:

- dynamic : intensity profile characterizing the sound.
- attack : a determining perceptive role to characterize the object.
- *allure* : characteristic variation in the sustain of some objects.

Finally, two variation criteria are also introduced :

- melodic profile : overall variations of the sound mass.
- mass profile : small scale variations of the sound mass.

For each of these criteria, Chion (and Schaeffer) provide a class or profile typology. This typology can partially be adapted to a computational approach. Some previous works already investigated this approach [Peeters and Deruty, 2008]. Ricard and Herrera [Ricard and Herrera, 2004] also confirm the relevance of this approach. Nevertheless, both studies also underly two main limitations. First of all, this typology is linked to a specific listening, namely *reduced listening*. It is an attempt to describe sound objects for what they are, regardless of their origin or influence on the listener. But this listening is not evident and can easily lead to confusions for an untrained subject. The other commonly underlined limitation is connected to the idea that new dimensions could be investigated in order to describe some sounds more precisely.

2.2 Temporal Semiotic Units and Parametric Temporal Patterns

Along the same lines, a formalism has been developed, over several years, by researchers at the *Laboratoire Musique et Informatique de Marseille* (Music and Computer Sciences Labs of Marseille). The *Unités Sémiotiques Temporelles* (Temporal Semiotic Units) [Delatour, 2005] provide a framework for musical analysis, specifically addressing the issue of musical semiotics, sketching out a general time semiotics in an artistic context. On the basis of 20 units, corresponding to distinct archetypes of sound dynamics and morphology, one can analyze musical pieces of various natures.

Lately, a mathematical and formal approach to these *UST* was developped with the *Motifs Temporels Paramétrés* (Parametric Temporal Patterns) [Bootz and Hautbois, 2005]. *MTP* describe the semantics of *UST* using graphic and analytic representations. Thus temporal profiles are drawn to describe an audio stream morphology. These profiles are built as an association of elementary functions, called *profilèmes*. These units display the same granularity as the ones we will model in this work.

Despite the humility of their authors on the validity of their work, the typologies described by Schaeffer or Chion, and the *UST* have a broad and general thrust. In this work, we want to inject less a priori knowledge on the studied materials. As a consequence, our paradigm is quite different. However, both approaches are not totally exclusive and one may try, with

the system that will be further described, to build object models relying on these typologies.

Chapter 3

Using Segmental Models

Among the different signal models, one may distinguish two main families: deterministic and statistic. Deterministic models tend to focus on specific known properties of the signal. For example, given the knowledge of a signal being composed as a sum of sine waves, one can estimate parameters like the amplitudes, frequencies and phases to build a model of this signal.

On the one hand, considering the variety of signals we must model in this work, such invariant properties can not be used a priori. On the other hand, the class of statistic models has been more deeply investigated for this study, as their flexibility allows us to model some variability. The goal of such a model is to characterize some statistical properties of the signal. It is assumed that real signals can be well modeled as realizations of parametric random processes where the parameters can be effectively estimated.

3.1 Markovian Models

In order to model the temporal development of observation series, Markovian models provide interesting formalisms and tools. We will thus study how these models can fit in our context.

3.1.1 Standard HMM Formalism

Lawrence R. Rabiner addresses most of the usual problems linked with the use of Hidden Markov Models in his tutorial[Rabiner, 1989]. First of all, Markov processes offer a modeling formalism for systems that can be fully described, at any time, with a finite number of different states. Each of those states corresponds to an observable event.

With HMMs, each state corresponds to a probabilistic function. It has proven to be useful to model systems where the events are not directly observable (they are hidden) but can be approached through observations resulting from some stochastic processes. An HMM is characterized by:

- a number of states N, which are hidden,
- a number of distinct observable symbols M,
- a probability distribution for state transition $A = \{a_{ij}\}$ with $1 \leq i, j \leq N$ where a_{ij} is the probability to go from state i to state j,
- a probability distribution for observation symbols given a state $B = \{b_i(k)\}$ with $1 \leq i \leq N$ and $1 \leq k \leq M$ where $b_i(k)$ is the probability to observe k if the system is in state i,
- an initial state distribution $\Pi = \{\pi_i\}$ with $1 \leq i \leq N$ where pi_i is the probability for the system to initialize with state i.

In the context of automatic recognition or segmentation, a basic problem is to find the optimal state sequence $Q = q_1 q_2 ... q_T$ that best explains a given sequence of observation $O = O_1 O_2 ... O_T$ (with a given set of parameters $\lambda = (\Pi, A, B)$). But the solution depends on how we define our optimization criterion. We may want to maximize the number of states that locally best explain the sequence, regardless of the likelihood of the global sequence. But if we want to maximize the posterior probability $P(Q|O, \lambda)$, the best path can be retrieved with Viterbi algorithm, defining the quantity:

$$\delta_t(i) = \max_{q_1, q_2, ..., q_{t-1}} P(q_1 q_2 ... q_t = i, O_1 O_2 ... O_t | \lambda) \tag{3.1}$$

which is the highest probability of a path ending in state S_i at time t. It can be computed recursively as:

$$\delta_{t+1}(j) = \max_i \ \delta_t(i)\ a_{ij}\ b_j(O_{t+1}) \tag{3.2}$$

$$\delta_1(j) = \pi_j\ b_j(O_1) \tag{3.3}$$

To explicitly get the state sequence, we must also keep track of the array $\phi_t(j)$ defined with:

$$\phi_1(j) = 0 \tag{3.4}$$

$$\phi_t(j) = \arg\max_i(\delta_{t-1}(i)\ a_{ij}) \tag{3.5}$$

Once these quantities have been evaluated for $1 \leq t \leq T$, the best state sequence $\hat{q_t}$ can be *backtracked* in this way:

$$\hat{q_T} = \arg\max_i \delta_T(i) \tag{3.6}$$

$$\hat{q_t} = \phi_{t+1}(\hat{q_{t+1}}), t = T-1, T-2, ..., 1 \tag{3.7}$$

3.1.2 Limitations

Standard HMMs offer interesting possibilities but also suffer from some weaknesses and limitations[Ostendorf et al., 1995]. We will now review some of the most important ones, with regards to our context.

Weak Duration Modeling

The major limitation of standard HMMs lies in duration modeling. Let us represent two states S_i and S_j of a larger HMM.

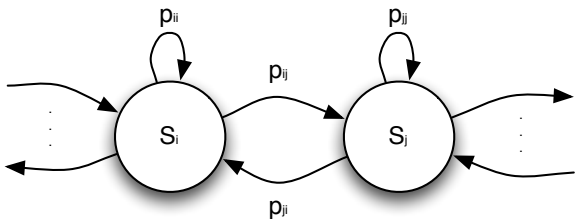

Figure 3.1: Two states of a larger HMM.

The probability that we get n consecutive observations in state S_i will then be given by :

$$p_i(n) = a_{ii}^{n-1}(1 - a_{ii}) \tag{3.8}$$

As a consequence, the standard HMM state duration model is given by a geometric distribution where the ratio is the auto-transition probability of the state. In our context, two segments referring to the same morphological class may not necessarily have the same length. As a consequence, we may want to provide an explicit duration model in order to address the duration flexibility our segments.

Conditional Independence of Observations

Another important limitation lies with the assumption of conditional independence of observations with standard HMMs. This problem has been studied and addressed in a large number of works[Bilmes, 1998]. An important element of this work is its focus on temporal evolution of successive observations. Therefore, conditional independence of observations is an assumption that would limit the robustness of our models.

Feature Extraction imposed by Frame-Based Observations

Last but not least, with standard HMMs, the observations are necessarily frame-based. When one wants to focus on temporal evolutions of a signal, it can be interesting to use descriptors that are not directly frame

related[Zue et al., 1989].
However, each state generates a single frame. Focusing on segments, subunities such as single frames are not necessarily meaningful. A segment-based model, with each state generating a variable-length sequence of frames, would thus allow us to emphasize the role of the elementary units we want to model.

3.2 Segmental Models

The limitations we pointed out are topics of concern for many different application fields. As a consequence, solutions have been given with many extensions of standard HMM models[Rabiner, 1989]. Among them, Segmental Models have acquired an increasing popularity in areas like speech processing or automatic handwriting recognition[Artieres et al., 2007]. We will thus investigate how well this extension can fit in our context.

3.2.1 How SM overcome HMM Limitations

The main idea, with Segmental Models, is to have each state (a in figure 3.2) generate a variable-length sequence of frames $(y_1, .., y_L)$ instead of a single frame y with standard HMMs, as this figure shows:

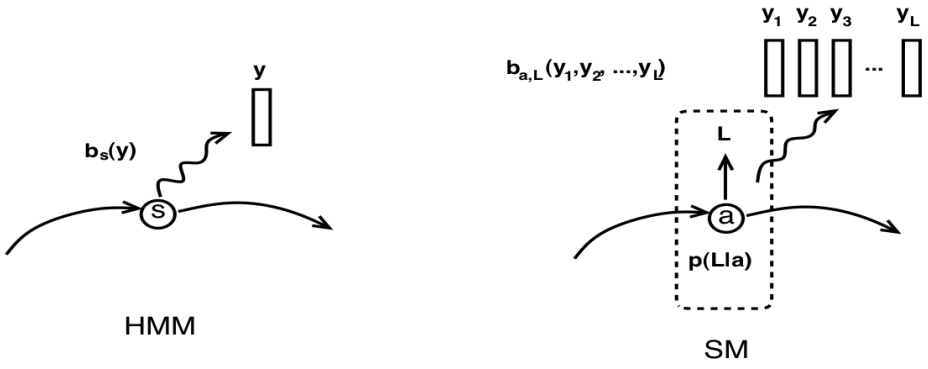

Figure 3.2: A frame-based HMM and a segment-based SM. (This figure has been copied from [Ostendorf et al., 1995])

As a consequence, this actually addresses all of the limitations we pointed out. At first, with this extension, each state is also characterized by an explicit duration distribution probability ($p(L|a)$ on figure 3.2). Secondly, each segmental state can generate an ordered sequence of frames. The assumption of conditional independence between observations is thus directly relaxed. Finally, we are obviously no longer restricted to using frame-based observation features and we can consider segment-based features.

3.2.2 From HMM to SM

As mentioned earlier, Segmental Models are an extension of standard HMMs. They can be considered as a generalization. As a consequence, the formalism remains similar. We'll now study how this generalization affects the results described earlier.

With standard HMMs, the observation distribution model is given by the following formula representing the probability of the state S_i generating the frame y:

$$b_i(y) = p(y|S_i) \tag{3.9}$$

With segment modeling, the observation distribution model represents the probability of the state S_i generating the frame sequence $y_1^L = (y_1, ..., y_L)$, given a length L:

$$b_{i,L}(y_1^L) = p(y_1^L|S_i, L) \tag{3.10}$$

As a generalization of standard HMMs, a similar procedure can be described with SMs to find the optimal state sequence $Q = q_1, ..., q_N$ that best explains a sequence of observation $O = O_1, ..., O_T$ (still with a given state of parameters). However, the introduction of the duration distribution for each state adds a dimension to the combinatory of the algorithm. This is why Viterbi, in the context of Segmental Models, is called *3D Viterbi*[Ostendorf et al., 1995].

The general framework is identical. However, with 3D Viterbi, equation 3.2 becomes:

$$\delta_t(j) = \max_i \max_{l \in \mathcal{L}} \delta_{t-l}(i)\ a_{ij}\ b_{j,l}(O_{t-l+1}^t)\ p_j(l) \tag{3.11}$$

In this new equation, one may notice the second max operator that is the consequence of the introduction of a third dimension. Beyond that, the algorithm remains similar. One slight difference: during the final backtracking step, we have to backtrack both $\hat{q}_t$ best state sequence and its associated duration sequence $\hat{d}_t$.

3.2.3 Model Parameters and how they can be set

The models we will use have been introduced, as well as the tools that will allow us to decode a sequence of observations as an optimal state sequence. However, these statistic models rely on a good determination or estimation of the parameters, and the results will dramatically depend on them. To make things clear, we will state here the set of parameters that have to be defined.

First of all, as described in 3.1.1, every HMM is characterized by:

- a number of states N and observation symbols M,
- a probability distribution for state transition $A = \{a_{ij}\}$, defining the topology of the model,
- an output probability distribution $B = \{b_{i,l}(k)\}$,
- an initial state distribution (often called prior) $\Pi = \{\pi_i\}$.

Moreover, as we are using SMs, each state of the model is also characterized by:

- an explicit duration probability distribution $p_i(l)$

Finally, with the use of 3D Viterbi algorithm, another parameter can be set:

- a global transition penalty α that will give priority to either longer or shorter segments during the decoding step (this issue is investigated in [Marukatat, 2004]).

For all these parameters, two different approaches can be considered. The first one consists in a true learning of the parameters. The idea is to learn, from annotated data, the best set of parameters through supervised (or semi-supervised) algorithms. However, this approach often leads to complex optimization problems. For example, [Rabiner, 1989] addresses the issue of finding the best set $\lambda = \{A, B, \Pi\}$ that maximizes $P(O|\lambda)$ given a sequence of observations O, with standard HMMs. In that case, there is no analytical solution to the exact problem. But the function can be locally maximized with classic algorithms such as *Expectation-Maximization* (EM). Automatic learning of the parameters aims at building a model that statistically best fits to the data. The drawback of this approach is that a large amount of labeled data is required.

Another approach consists in setting the parameters a priori. The choice of parameters can thus be driven by a priori knowledge of the signals to model or by some modeling choices. One important assumption behind this approach is that, given a specific task using the models, the set of parameters that achieve best results may be different from the optimal set of parameters that statistically best fit the data. For example, for some classification tasks a set of parameters that allows a better discriminability between the classes will often be more useful than a set that best generates the observed data.

Part II

Achieved Work

Chapter 4

Extensions and Implementation of the Model

The characteristics we want to describe, and the models we will use to do so, have now been introduced, in a general framework. We can now address the specific issues of this work. More specifically, we will now discuss the properties of the objects we will model, the observations we will work on and how the parameters of our models listed in 3.2.3 will actually be set.

4.1 The Elementary Objects and their Description Space

Potentially, the diversity of the objects we will model may be huge. However, the computational framework of this study sets some limits we have to define.

4.1.1 Properties of the Objects and Redefinition of the Segmentation Task

Within a morphologic class, the temporal evolution of the observations is the common characteristic. But this evolution does not depend on the global duration of the different occurrences of the class. As a consequence, we will have to study the similarities between objects with different lengths.

To do so, we will have to allow some temporal variability in our models. In figure 4.1, we can see that the two curves have strong similarities in their temporal development. However, assuming that we would like to define a morphological class that would include both these curves, it is impossible to deduce one occurrence from the other using a linear stretch. Some tools like *Dynamic Time Warping*[Keogh and Ratanamahatana, 2005] would allow us to overcome this problem and to realign the curves with non-linear stretches as a pre-step to build our models. But the computational complexity of this

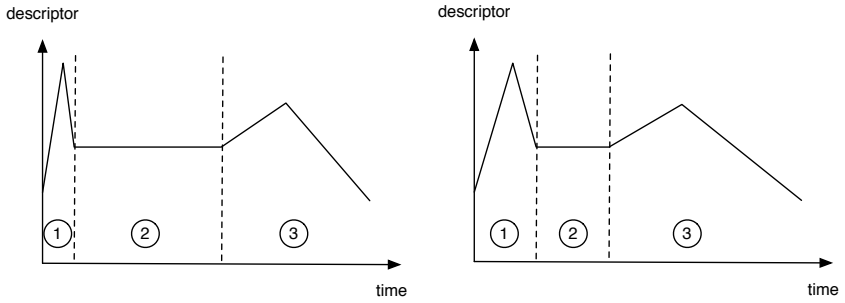

Figure 4.1: Two objects that can not directly be deduced from each other with a linear stretching.

step makes us investigate another way.

If one actually cuts the curves into 3 zones, as displayed on figure 4.1, the problem of retrieving one whole sequence from the other, using linear stretches can be solved. In fact, allowing only linear stretching does not reduce the generality of this work as long as we cut our segments in small enough parts.

As a result, the granularity of the objects we will model is now set. Our elementary segments will be defined by the following property: two segments referring to the same class can be temporally aligned by applying a linear stretch. The segmentation task can thus be described as a two-step work :

- Extraction of a dictionary of elementary objects,
- Given a dictionary, decoding the sequence of elements that will best explain the whole audio stream.

4.1.2 Nature of the Observations and Limits of the Study

Depending on the objects we want to model, the relevant description spaces will be different. To give a few examples, if one wants to build a model of glissando, pitch descriptors will be adapted. With crescendo models, loudness descriptors will fit better. On the contrary, using pitch descriptors to describe a crescendo would make no sense. We can assume that the choice of the descriptors will dramatically affect the quality of our models.

However, the evaluation of the quality of descriptors or the automatic selection of adapted descriptors is not an issue we want to address in this work. As a consequence we will use a predefined set of descriptors throughout this study. Every set of descriptors will have its own benefits and drawbacks. But a good compromise can be found using timbre descriptors. We will

thus use a set of 8 Mel Frequency Cepstral Coefficients (including the first one). These descriptors have proven to be particularly efficient for speech description but also for musical signals[Logan, 2000].

An important assumption is also to be made. We are using a multidimensional description space. As a consequence, the sequences to describe will also be. However, these dimensions will be considered as synchronous. In other words, the signal processing tasks will be done over all the dimensions at the same time. Segmenting separately along different sets of descriptors, and then using data fusion is an approach that will not be addressed here.

4.2 Setting the Model Parameters

For every model parameter listed in 3.2.3, we will now discuss the choices that we made and describe the methods used to estimate them.

4.2.1 The Number of States and their Topology

As we do not have a priori knowledge of the segments and the way they are organized, an ergodic topology will be given to the Markovian model. As a consequence, all transitions between states, as well as auto-transitions, will be allowed and equiprobable.

4.2.2 Output and Duration Probability Distribution

Though complex output distributions and conditional probabilities between successive observed frames can be examined, the most simple assumptions will be done with respect to the small body of annotated data we will have. If one considers a single output distribution and independent and identically distributed successive frames, the output distribution model is:

$$b_{i,l}(y_1^l) = \prod_{j=1}^{l} p(y_j|i) \tag{4.1}$$

One may notice that a segmental state generating a sequence y_1^L is then analogous to a pure left-right HMM with L states, each generation a frame y_i:

For each segmental state, we have to define an explicit output distribution model. We will assume that a state generates a mean sequence with a deviation modeled like a sequence of centered gaussian noises. Let us consider a sequence of observations θ_i, we have:

$$\theta_i = \hat{\theta}_i + b_i \tag{4.2}$$

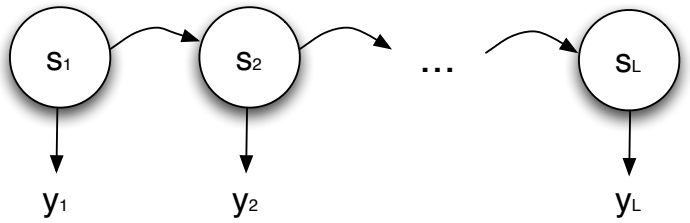

Figure 4.2: The equivalent left-right semi-Markov model.

where $\hat{\theta}_i$ is the mean sequence and b_i is the gaussian noise. As a consequence, this mean sequence curve $\hat{\theta}_i$ will have to be estimated, as well as the parameters of the gaussian noises b_i.

Estimating the Mean Sequence

For each segmental state, we can estimate the parameters from annotated occurrences of the related class. These occurrences refer to a common morphological class and thus share some similarities in their temporal behavior. However, their lengths are different. It is thus essential to stretch the occurrences so that they can fit within the boundaries set in 4.1.1, as we see in figure 4.3.

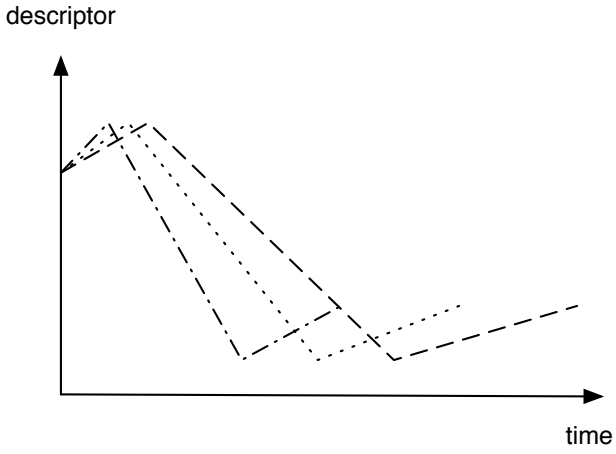

Figure 4.3: Three occurrences of an exact identical trajectory with different lengths.

Moreover, with actual data, the curves generally do not directly match with a simple resampling to the same length. A temporal alignment is necessary in most of the cases, as illustrated in figure 4.4.

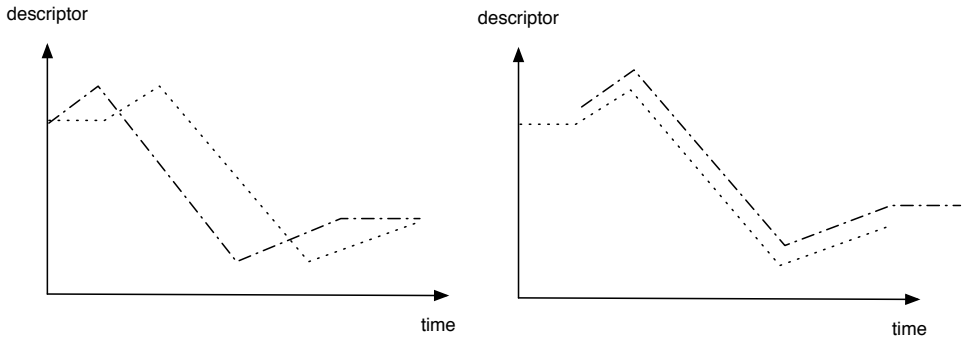

Figure 4.4: Two occurrences with a same length but that need an alignment as displayed on the right graph.

The first important step thus consists in aligning theses occurrences. In this work, we will just consider linear stretching in order to align curves. Then, the problem boils down to finding the best combination of stretching and time lag that maximizes the cross-correlation between the two curves.

Once all the occurrences have been properly aligned, we can estimate a mean trajectory. However, even if the curves are well aligned, the mean estimated trajectory may not locally capture the dynamics of the real signals, as figure 4.5 shows.

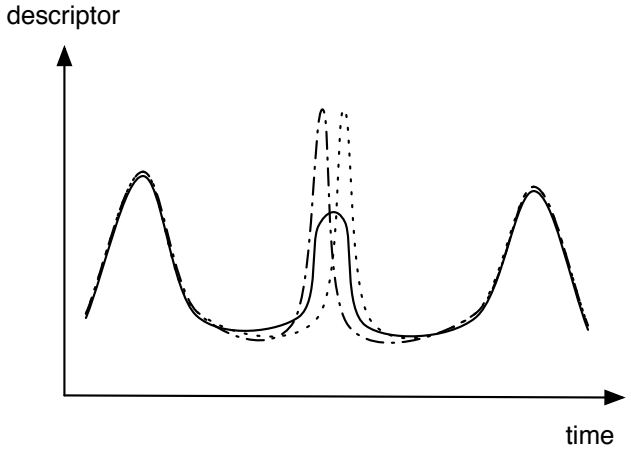

Figure 4.5: Two properly aligned occurrences (dotted lines) with their mean estimated trajectory (plain).

To address this problem, one of the solutions is to add another step. Using the mean estimated trajectory in the model, we can choose among the

occurrences the one that would be most likely generated. Then, we can use this most likely (and real) trajectory for our model. The differences between the two built state models will be further investigated.

The Gaussian Noises Variances

We are considering multi-dimensionnal segments. Let us define D as the number of dimensions of the observation vectors and L as the length of the segment (which is the number of temporal frames). Assuming the independence of our successive observation frames, we will have to estimate the parameters of the L-length sequence of multivariate gaussians. Each of these gaussians will theoretically be characterized by a $D \times D$ covariance matrix.

In order to reduce the number of parameters, we will assume that the different dimensions of our signal descriptors are statistically independent. As mentioned earlier, the system and modeling choices do not depend on the chosen descriptors. And as the description space may consist of drastically different dimensions (audio descriptors combined to physical gesture capture recordings, for example), this assumption is not unrealistic .

The covariance matrix of our gaussians then becomes diagonal and there are D coefficients to estimate for each gaussian. This leaves us with $D \times L$ parameters to estimate for the whole segmental model. One may notice that these D-multivariate gaussians are thus equivalent to D univariate gaussians.

We could still reduce this number of coefficients. However, keeping the $D \times L$ coefficients allows us to keep a precise control over the deviation tolerance of our models. For example, in order to model a specific musical event, we may want to restrict deviation on some critical temporal parts while a larger deviation in other portions may not drastically change the perception of the segment. This is illustrated with figure 4.6 where the different realizations of the model have a similar profile, and a deviation tolerance that is not constant through time.

In the same way, the deviation tolerance can be independently controlled for each dimension of the description space. Practically, each coefficient will be computed with regards to the aligned occurrences and the mean trajectory.

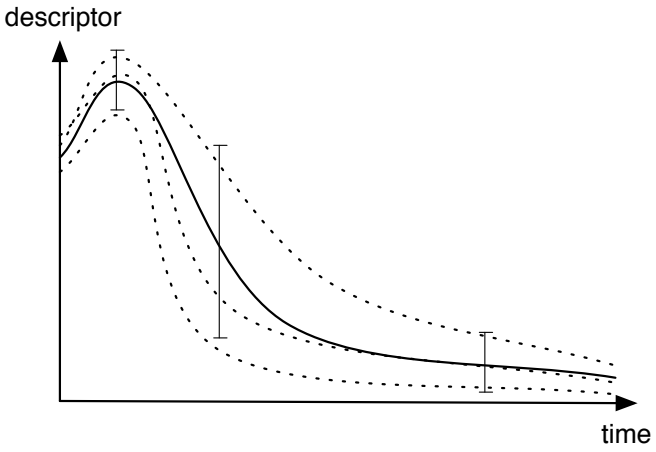

Figure 4.6: An unidimensional model (with its center displayed as plain strokes) with non-constant variance and three of its realizations (displayed as dotted lines).

The Duration Probability Distribution

Many options could also be considered to model the duration probability distribution function. It was decided to set the limits of a range of allowed durations and to assume that the different values between theses boundaries were equiprobable.
The boundaries of the allowed duration range still have to be chosen. For the lower boundary, the value is less important than the length of the shortest occurrence. Precisely 0.7 times this shortest length will be kept as the first boundary. In the same way, 1.3 times the longest length will be chosen for the second boundary.

4.2.3 3D Viterbi Algorithm

Finally, contributions were also made within the 3D Viterbi algorithm. More specifically, one improvement has been made for the backtracking step. It may happen that the sequence we want to decode does not finish exactly on the end of the realization of a state segment. As a consequence, the best state sequence will be obtained through backtracking from $O_{\hat{T}}$, with $\hat{T}$ chosen to be the end time maximizing the global likelihood of the decoded sequence:

$$\hat{T} = \underset{T-L+1 \leq t \leq T}{\arg\max} \max_{j} \delta_t(j) \tag{4.3}$$

where L is a quantity representing the longest duration that may not be segmented at the end of the observation sequence.

Moreover, as mentioned earlier, the decoding involves the computation of the likelihood of a state S_k given an observed sequence $\hat{y_1}, ..., \hat{y_L}$. With mono-dimensional signals, in the logarithmic domain, this likelihood can be approximated with the following formula[Artieres et al., 2007]:

$$\log\ p(\hat{y_1}, ..., \hat{y_L}|L,\ S_k) = \frac{-1}{2}\ \sum_{i=1}^{L} \frac{(\hat{y_i} - y_i)^2}{\sigma^2} \tag{4.4}$$

As we are working with multi-dimensional signals, considering that our models and the assumed independence between dimensions, we will use the following formula:

$$\log\ p(\hat{y_1}, ..., \hat{y_L}|L,\ S_k) = \frac{-1}{2\ D\ L}\ \sum_{i=1}^{L} \sum_{j=1}^{D} \frac{(\hat{y_ij} - y_{ij})^2}{\sigma_{ij}^2} \tag{4.5}$$

where $\hat{y_i}j$ is the j-ith coordinate of the i-ith observation vector in the sequence. Moreover, the $\frac{1}{DL}$ factor will normalize the likelihood with respect to the number of dimensions and the length of the sequence.

Chapter 5

User Interaction Model

So far, we have investigated how models could be built to represent the temporal evolution of signals and how a segmentation relying on these characteristics could be computed. The last aspect we wanted to address was the methods we could use to integrate user-specific prior knowledge and preferences in the learning process. We will assume that this can be done through an interaction loop between the system and the user. We will now describe the use case of this particular work and the interaction paradigm that is investigated. This chapter introduces the possible building blocks of a user-interaction-based learning algorithm. Although not fully automated in our system, we carried on with this learning paradigm through the rest of our work.

5.1 Paradigm and Use Case

The interaction between a system and its users can take many forms. In order to enhance the results that could be obtained with a fully automatic unsupervised segmentation task, the supervision from an expert user can be of great help [Chapelle et al., 2006]. However, for this supervision to be fruitful, the roles of the different agents have to be clearly depicted.

In the prototype system we implemented, the user will first provide a partial segmentation of the stream where occurrences of the different morphological classes will be marked and labeled. From these data, models of the different classes will be built as described in chapter 4. Using these models an exhaustive segmentation will be proposed. Depending on his satisfaction, the user will either validate the segmentation or try to improve it. If so, some more precise feedback will have to be given so that the system can generate a segmentation that will eventually satisfy the user. This process will be iterated and an interaction loop will be set between the system and the user.

The system is deterministic in the sense that given a stream to segment and a set of annotations, used as an input, the system will always derive the same models, and generate the same segmentation. As a consequence, to obtain different output results, a different annotation input has to be provided. Various modifications in this input can be considered. Given a class, fewer or more of its occurrences can be manually labeled. New classes that had not been considered in the first place can be defined. If a class contained occurrences that were too different, it could be divided into to separate classes. On the contrary, two classes that were too similar could be grouped into a single one.

The reasons for a divergence between the segmentation provided by the system and a satisfying solution from the user perspective can be multiple. We will further study some of the main ones. But the basic idea in this interaction is to have the system try to point out the problems and identify their origins. Instead of trying to have the system solve all the problems itself, the expert user will be asked to guide the algorithm.

5.2 The Main Problems and their Identification

In the interaction loop between the system and the user, the state models are built in a learning step. From that point, a segmentation is proposed given the state models, in an inference step. These two steps are opportunities to detect different problems. For both, we will now investigate the main difficulties that may be identified and methods to automatically detect them.

5.2.1 Class Modeling Issues

Case of a class with a too high tolerance to variation

If one tries to model a class allowing far more variability than the others, the σ_{ij} coefficients will globally be assigned higher values. Considering equation 4.5, one can notice that given an observation sequence and a state mean sequence, the likelihood of the observed sequence being generated by the state model will be all the more important since σ_{ij} coefficients are large.

As a consequence, a sequence corresponding to a given morphological class might as well be more likely be generated by another state model, as long as it has a higher variability tolerance. This situation can thus lead to misclassification problems. Fortunately, this issue can be addressed by

computing, for the state models, the mean variance coefficients :

$$\bar{\sigma} = \frac{1}{LD} \sum_{i=1}^{L} \sum_{j=1}^{D} \sigma_{ij} \tag{5.1}$$

If one of these coefficients is significantly higher than the others, the user will be warned that the corresponding class may refer to occurrences that are morphologically too different. Moreover, the origins of this problem can also be identified. Once the state models have been built, the likelihood of each of the occurrences being generated by their associated state model is computed. The lowest likelihood will be attained for the outliers of each class. This information will also be given to the expert user.

Ambiguity between States

If two different classes describe too similar temporal evolutions, from the system perspective, there may be some ambiguity between the state models. This ambiguity is also a common source of misclassification. A divergence measure between the different state models can give an insight of the ambiguity of the models.

In information and probability theory, this issue has been widely investigated [Burbea and Rao, 1982] [Basseville, 1988]. Among the large number of existing similarity or divergence measures, the Kullback-Leibler divergence measure [Kullback and Leibler, 1951] was chosen. For probability distributions P and Q of a discrete random variable, the K–L divergence of Q from P is defined as :

$$D_{KL}(P||Q) = \sum_{i} P(i) \ \log \frac{P(i)}{Q(i)} \tag{5.2}$$

This measure is not symmetric but can be symmetrised through the computation of :

$$D_{KL}(P||Q) + D_{KL}(Q||P) \tag{5.3}$$

One can notice that this actually is a pseudo-distance as the symmetry, non-negativity and identity of indiscernibles properties are satisfied.

If the relative distance between two states output probability distributions is low, this may indicate that the class definitions may be ambiguous.

5.2.2 Case of Portions of the Stream that can not be explained with the Modeled Classes

The whole segmentation process relies on the identification of recurring shapes in a stream. But often, the stream to segment may contain singular shapes that do not match any of the morphological classes. However,

the Viterbi algorithm will still try to decode the full sequence and try to find a combination of states that best explains the singular shape.

Once the best path has been computed and backtracked, for each state of the sequence, the likelihood of each segment being generated by the retrieved state model will be computed. Spotting the regions where the likelihood values are significantly low is a good way to detect this problem.

In order to overcome the segmentation problems related to this issue, a special label is set to let the system know that it should not try to segment some part of the stream. If one considers a whole sequence $y_1, ..., y_T$ and one region delimited by two indexes l_1 and l_2 (so that $1 \leq l_1 < l_2 \leq T$) that one does not want to automatically segment, the segmentation task will then be divided into two parts. The Viterbi algorithm will independently be used on $y_1, ..., y_{l_1-1}$ and $y_{l_2+1}, ..., y_T$.

Doing so will actually improve the overall results as some knowledge has been implicitly transfered to the system. Indeed, the assumption that a segment will end with y_{l_1-1} and that another one will start with y_{l_2+1} is done.

Chapter 6

Results

In this chapter, we will introduce the most important results that have been obtained. We will particularly focus on three types of sound materials that were used. Each of these examples have uncovered some major aspects and issues of our approach.

6.1 The Elephant

In order to validate the first implementation of the global system, a simple segmentation task was tested and evaluated.

6.1.1 Description of the Data

The first series of materials we worked with consists of recording of environmental sounds. At different moments, the audio environment of the *Place Stravinsky*, by the entrance of IRCAM, was recorded by our colleague Diemo Schwarz. More specifically, the characteristic sounds produced by the elephant of the *Stravinsky Fountain* were captured in one set of recordings lasting between 40' and 5 minutes approximatively, at different times of the day.

Actually, the elephant is a mechanical automat. Periodically, the elephant belches water through its trunk. Due to the rust and greasing of the mechanisms, this action produces a characteristic squeak with a noticeable dynamics. Within a recording, this sound event is reproduced with some small variations in its "pitch" and length. Thus, the sound produced by the elephant is a perfect object for a first segmentation task, using real data.

For some reason, depending on the moment when the recordings were made, the squeak was somewhat different but still, all of them shared a common dynamics.

Figure 6.1: The elephant of the Stravinsky Fountain

6.1.2 Segmentation and Results

In a first attempt, two classes were chosen to model the data. The first one consisted of the squeak sound of the elephant (labeled as 2). The second one was the environment sounds between two squeaks (labeled as 1). We can hear the flow of the fountain water and sounds coming from the surrounding area (people talking mainly).

In order to evaluate the quality of the automatic segmentations, a manual segmentation was made to serve as a reference. It consists of sequences of classes 1 and 2 occurrences as displayed on figure 6.3. The first task consisted in learning models with little learning data (2 occurrences of each class) and performing the segmentation on the same stream.

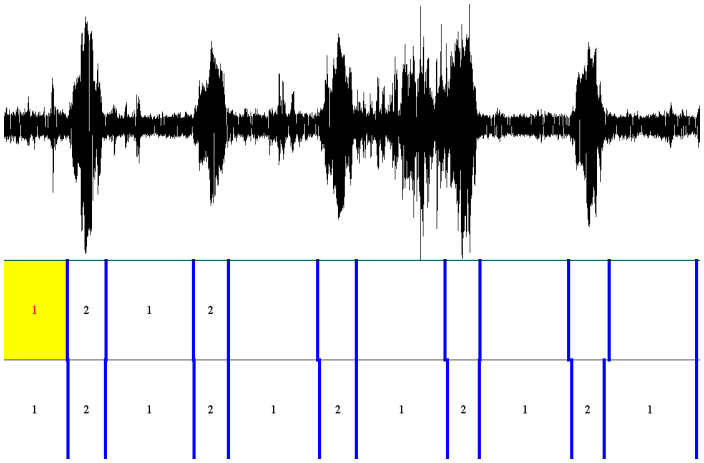

Figure 6.2: An excerpt of the first stream with its wavefront and two segmentations (the reference one on the top and the result at the bottom).

In this first task, all the occurrences of squeaking sounds were segmented and labeled correctly. If we compare the resulting segmentation with the reference one, frame by frame, we can claim we obtained almost 100% correct classification. Allowing a temporal tolerance of 100ms for the beginning and end markers of each segment, 94% segments can be considered as segmented correctly. Considering a tolerance of 400ms (20% of the shortest segment duration), we obtained 100% of correct segmentation and classification. Apart from the divergence of the marker positions, all segments matched perfectly with the reference segmentation.

Paying closer attention to the precise position of the markers, in many cases, we found that the automatic segmentation is even better than the manual one. Indeed the dynamic captured by the model is such that the events are segmented in a consistent way and there is a strong regularity in the position of the start and end markers.

Using the same stream, a second segmentation task was performed. As mentioned earlier, the squeaks were slightly different along the recording. But if one focuses on the evolution of the "pitch", one could divide the squeaking sounds into two classes.

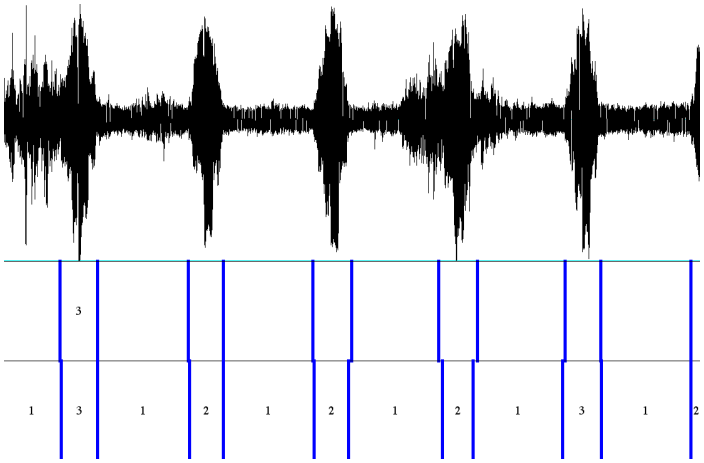

Figure 6.3: Another excerpt of the first stream with a third class corresponding to a specific squeak type.

Along the whole stream, 15 squeaks can be identified. In this task, the 15 squeaks were precisely segmented. Moreover, except for one occurrence, the squeaks were correctly classified in the 2 classes of squeaking sounds.

In a last segmentation task, we used the models built in the first task to segment another stream taken from a different recording. Despite the differ-

ences between the occurrences used to train our models and the ones that could be identified in the segmented stream, the system could still perform the task without error.

These first segmentation tasks were rather simple but provide a first validation of the proposed algorithm. The system can actually retrieve "real" audio objects that are morphologically similar but have some variability in their temporal development and length. These results are all the more satisfying than the cepstral description space is not optimal to capture the evolution of the perceived pitch of the squeaking sounds.

6.2 Concatenation of Fixed Sounds

Some further experiments were conducted with manually-concatenated audio streams. This time the different occurrences in the stream were exact copies with no variation. But this was the opportunity to study the impact of the learning data on the results more intensively.

6.2.1 Description of the Data

During an interview with the composer Roque Rivas, we could learn about his particular strategies and criteria he applies to the segmentation of sound materials in his work. In a meticulous way, Rivas collects a lot of various audio materials and organizes their storage using a complex typology. For some materials, the classification criteria for his database were very close to the morphological criteria we are interested in.

In his piece *Conical intersect*, Rivas mixes bassoon recordings with various processed environmental sounds and vocal performances. Among his huge collection of environmental sounds, we selected four samples. Two of them were sounds produced by the ignition of canons and the two others consisted of paper being torn.

6.2.2 Segmentation and Results

The first task consisted in building four classes corresponding to the four types of samples. As the occurrences in the stream were exact copies, this task was extremely simple and the system performed an exact and precise segmentation.

For the second task, only two classes were built. One class corresponded to the canon sounds, and the second class regrouped the paper sounds. For each class, the models were built using both samples. This task was hardly

more difficult and the system achieved 100% correct classification with an imprecision on the marker position below the duration of the analysis window.

The last task was performed using the same two classes. But this time, the classes were built using only one type of samples from each class. As we can see in figure 6.4, the canon events (label "2") were segmented as expected. But unexpected results were obtained for the paper events (label "1"). For the sample used as learning data (corresponding to label "1a" in the reference segmentation), expected results were obtained. But surprisingly, the system considered the second paper sample (label "1b") as a sequence of two occurrences of the paper class.

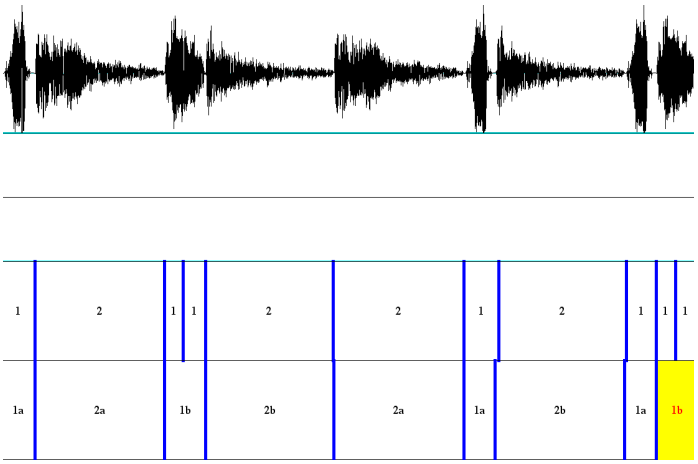

Figure 6.4: The segmentation result on top and the reference at bottom

From a semantic point of view, the first paper sound is similar to the second one, as they refer to the same action. But when one abandons this perspective and pays more attention to the temporal development of the two samples, the results given by the system actually make more sense.

Beyond the pleasant surprise of the system giving unexpected but meaningful results, these results highlight a major difficulty of evaluating the quality of the results. Indeed, a basic comparison between the resulting and reference segmentation would lead to the conclusion that the system made some mistakes. But the segmentation the system provides is at least as valid as the reference. As a consequence this result directly calls into question the very existence of a unique reference segmentation.

If we cannot rely on a reference to compare our results, the usual evaluation methods do not apply to our context.

6.3 A Real Musical Piece

During the four months that this internship lasted, the reflections on many issues were fed through the exchange with composers. Among these meetings, those with the composer Marco Antonio Suárez-Cifuentes were particularly fruitful. Suárez-Cifuentes pointed out that the temporal evolutions of some properties of the signal was actually one of the most important aspects to take into account in the selection and use of audio material in his composition work. As a consequence, working with some of his pieces was a valuable opportunity to test the system in the conditions that motivated this study.

6.3.1 Description of the Data

Most of the work with Suárez-Cifuentes' data was done with an extract taken from his latest piece *Poetry for // dark -/ dolls*. This extract is a solo performance using various vocal techniques. Throughout the piece, recurrent timbral evolutions can be identified. As a consequence, this piece constituted a stream of choice for this study.

However, given the complexity of the piece, choosing the partial segmentation to give as an input to the system was a critical and difficult task. In those conditions, the idea of having the composer himself provide this input made sense. Watching the composer performing this task actually provided a lot of insight on what a professional user could expect from a segmentation tool.

With this set of data, many more different classes were identified compared to the materials that had been used so far. The variability of those classes was also far greater than what we had previously considered.

6.3.2 Segmentation and Results

In an iterative and interactive process, three steps were made in order to segment this stream. We will now show the results of these consecutive iterations.

On the part displayed in figure 6.5, the results, though not erratic, are not satisfying. One of the main problems is that a lot of segments are labeled as occurrences of the fifth class. Actually, this is due to the definition of this class. As described in 5.2.1, the variability among the occurrences that were given to build this state model is too important.

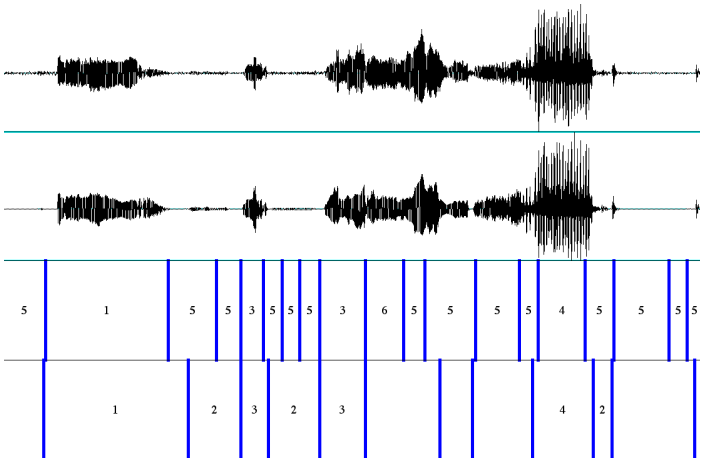

Figure 6.5: The first attempt (the segmentation result on top and the input at bottom)

Taking this useful information into account, Marco provided a new input. Although the results were improved, we still had to face a number of similar problems. Finally, a last attempt was made with a third input.

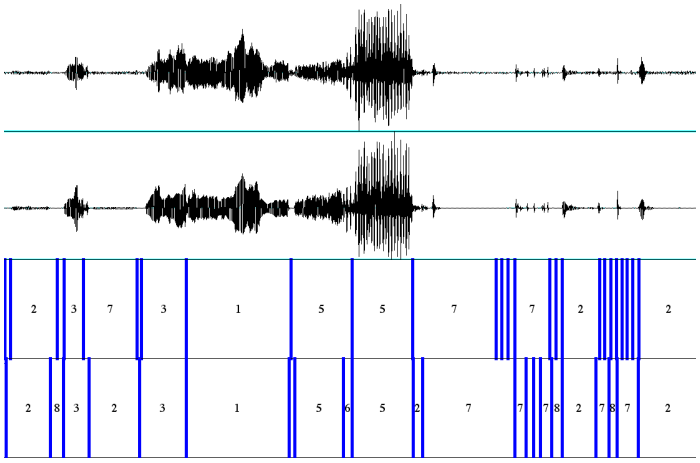

Figure 6.6: The third attempt (the segmentation result on top and the input at bottom)

Ultimately the results partially displayed on 6.6 were significantly better than the ones obtained with the first iterations. Beyond that, one can notice that between the first and last iterations, the input given by Marco really evolved. More classes have been identified in the process. Moreover, on may notice that globally, the segments Marco provided are far shorter. This phenomenon can be interpreted as a progression in the extraction of

the elementary units constituting the whole stream.

Even though very encouraging, the results are not yet directly usable for applications like concatenative synthesis. In the process, one can observe that the interaction between the user and the system was really fruitful. The input given by the composer influenced the output of the system. But the output of the system also influenced the next inputs that the composer provided.

From a totally qualitative point of view, one may also add that the feedback by Suárez-Cifuentes about the system was very positive. More specifically, he underlined that the results were actually improving in the process, from his point of view. In addition, he also expressed enthusiasm when the system revealed solutions that he had not expected.

Chapter 7

Conclusion

As a conclusion, the achieved work in this study can be briefly summed up. Here are the main achievements that one may particularly highlight.

In the first place, a prototype of a semi-supervised and interactive segmentation tool was implemented. With this prototype, the user provides a partial annotation of the stream he wants to segment. In an interactive loop, the system is able to build models of the morphological classes the user defines. These models will then be used to provide an exhaustive segmentation of the stream, generalizing the annotation of the user.

This achievement relies on the use of Segmental Models, that have been adapted and implemented for sound streams represented by a set of audio descriptors (MFCC). The very novelty with this study is to use real data to build models of the morphological classes, issued from various audio materials. A singular method to build our global model is defined, using both learning paradigms and the integration of user knowledge.

The global approach of this work is validated through experimentations with both synthesized streams and real-world materials (environmental sounds and music pieces). A qualitative and less formal validation also emerges from the feedback given by composers that worked with us along the whole internship.

Chapter 8

Future Directions

Segmental Models have proven to be effective in this study in a generative processing. For the segmentation task, it may be interesting to investigate the use of discriminative models. Considering our context, Conditional Random Fields [Lafferty et al., 2001] may provide an adequate extension of the developed approach.

In order to enhance the results achieved through the interactive process, the iterative process has to be formalized with more clarity.. More specifically, we have to precisely study the criteria and conditions of a convergence between the segmentation suggested by the system and the expectations of the user.

Another aspect that limits the interaction is the computation time for each iteration. The 3D Viterbi algorithm, in its current implementation (Matlab, without approximation) is too heavy to ensure that a segmentation can be computed within a decent time. The fluidity of the workflow, alternating user annotation and the feedback by the system, depends on the time the system needs to compute one iteration. An optimization of the algorithm, with approximations of the best solution, such as pruning methods [Ostendorf et al., 1995] shall then be explored.

An extension of our system could also be implemented to process real-time audio streams with a pre-trained system. The classic Viterbi algorithms, relying on posterior probability computations, is not meant for real-time. However, extensions, like Short-Time Viterbi [Bloit and Rodet, 2008], can overcome this problem at the cost of an approximation of the optimal solution.

We did not want to address the issue of an automatic feature selection in this work. However, the selection of an adapated description space could

increase the quality of the segmentation task. This aspect could thus be explored, taking into account the specificity of our interactive approach.

In this first prototype, we only investigated ergodic topologies for our model. To model a specific sequencing, other topologies could be considered. In order to model the organization of the elementary units inside the stream, a hierarchical model could be used. A multi-layer segmentation would then be performed, allowing us to access the different semantic layers of the materials.

Last but not least, such a segmentation system could be a valuable contribution to the existing applications based on segmented sound materials. Corpus-based concatenative synthesis, automatic improvisation system such as OMax [Bloch et al., 2008], score and gesture following tools as well as automatic orchestration may benefit from a system like the one we proposed and prototyped in this work.

Bibliography

[Artieres et al., 2007] Artieres, T., Marukatat, S., and Gallinari, P. (2007). Online handwritten shape recognition using segmental hidden markov models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29:205–217.

[Basseville, 1988] Basseville, M. (1988). Distance measure for signal processing and pattern recognition. Technical report, INRIA.

[Bilmes, 1998] Bilmes, J. A. (1998). Data-driven extensions to hmm statistical dependencies. In *Proceedings of International Conference on Spoken Language Processing*, pages 69–72, Sidney, Australia.

[Bloch et al., 2008] Bloch, G., Dubnov, S., and Assayag, G. (2008). Introducing spectral and video features into the omax improvisation system. In *Proceedings of International Computer Music Conference*, Belfast, United Kingdom.

[Bloit and Rodet, 2008] Bloit, J. and Rodet, X. (2008). Short-time viterbi for online hmm decoding : evaluation on a real-time phone recognition task. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, Las Vegas, Nevada, USA.

[Bootz and Hautbois, 2005] Bootz, P. and Hautbois, X. (2005). Les motifs temporels paramétrés. In *Vers une sémiotique générale du temps dans les arts*, Paris, France.

[Burbea and Rao, 1982] Burbea, J. and Rao, C. (1982). Entropy differential metric, distance and divergence measures in probability spaces: A unified approach. *Journal of Multivariate Analysis*, 12-4:575–596.

[Chapelle et al., 2006] Chapelle, O., Schölkopf, B., and Zien, A. (2006). *Semi-Supervised Learning*. The MIT Press.

[Chion, 1983] Chion, M. (1983). *Guide des Objets Sonores*. Buchet/Chastel.

[Collins, 2002] Collins, N. (2002). The bbcut library. In *Proceedings of the International Computer Music Conference*, pages 16–21, Göteborg, Sweden.

[Delatour, 2005] Delatour, I. ., editor (2005). *Vers une sémiotique générale du temps dans les arts*, Recherche et Création Musicales, Musique / Sciences, Marseille, France.

[Duxbury et al., 2003] Duxbury, C., Bello, J., Davies, M., and Sandler, M. (2003). A combined phase and amplitude based approach to onset detection for audio segmentation. In *Proceedings of the 4th European Workshop on Image Analysis for Multimedia Interactive Services*.

[Gouyon and Herrera, 2003] Gouyon, F. and Herrera, P. (2003). A beat induction method for musical audio signals. In *Proceedings 4th WIAMIS Special Session on Audio Segmentation and Digital Music*, London, England.

[Jehan, 2005] Jehan, T. (2005). *Creating Music by Listening*. PhD thesis, Massachusetts Institute of Technology.

[Keogh and Ratanamahatana, 2005] Keogh, E. and Ratanamahatana, C. A. (2005). Exact indexing of dynamic time warping. *Knowledge and Information Systems*, 7-3:358–386.

[Klapuri, 1997] Klapuri, A. (1997). *Automatic transcription of music*. PhD thesis, Tampere University of Technology.

[Kullback and Leibler, 1951] Kullback, S. and Leibler, R. (1951). On information and sufficiency. *The Annals of Mathematical Statistics*, 22-1:79–86.

[Lafferty et al., 2001] Lafferty, J. D., McCallum, A., and Pereira, F. C. N. (2001). Conditional random fields: Probabilistic models for segmenting and labeling sequence data. In *Proceedings of the Eighteenth International Conference on Machine Learning*, pages 282–289, Williamstown, Massachusetts, USA.

[Lindeman, 2007] Lindeman, E. (2007). Music synthesis with reconstructive phrase modeling. *Signal Processing Magazine, IEEE*, 24-2:80–91.

[Logan, 2000] Logan, B. (2000). Mel frequency cepstral coefficients for music modeling. In *Proceedings of the International Symposium on Music Information Retrieval*, Plymouth, Massachusetts, USA.

[Lu et al., 2002] Lu, L., Zhang, H.-J., and Jiang, H. (2002). Content analysis for audio classification and segmentation. *IEEE Transactions on Speech and Audio Processing*, 10:504–516.

[Marukatat, 2004] Marukatat, S. (2004). *Une approche générique pour la reconaissance de signaux écrits en ligne*. PhD thesis, Université Pierre et Marie Curie (Paris-6).

[Orio et al., 2003] Orio, N., Lemouton, S., Schwarz, D., and Schnell, N. (2003). Score following: State of the art and new developments. In *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, Montreal, Canada.

[Ostendorf et al., 1995] Ostendorf, M., Digalakis, V., and Kimball, O. (1995). From hmm's to segment models: A unified view of stochastic modeling for speech recognition. *IEEE Transactions on Speech and Audio Processing*, 4:360–378.

[Peeters, 2004] Peeters, G. (2004). Deriving musical structure from signal analysis for music audio summary generation : "sequence "and "state "approach. In *Lecture Notes in Computer Science*, volume 2771, pages 143–166. Springer-Verlag.

[Peeters and Deruty, 2008] Peeters, G. and Deruty, E. (2008). Automatic morphological description of sounds. In *Proceedings of Acoustics'08*, Paris, France.

[Rabiner, 1989] Rabiner, L. R. (1989). A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, pages 257–286.

[Ricard and Herrera, 2004] Ricard, J. and Herrera, P. (2004). Morphological sound description : computational model and usability evaluation. In *Proceedings of the Audio Engineering Society : 116th Convention*, Berlin, Germany.

[Schaeffer, 1966] Schaeffer, P. (1966). *Traité des Objets Musicaux*. Seuil.

[Schwarz, 2007] Schwarz, D. (2007). Corpus-based concatenative synthesis: Assembling sounds by content-based selection of units from large sound databases. *IEEE Signal Processing*, 24-2:92–104.

[Zue et al., 1989] Zue, V., Glass, J., Philips, M., and Seneff, S. (1989). Acoustic segmentation and phonetic classification in the summit system. In *Proceedings of International Conference on Acoustics, Speech, and Signal Processing*, pages 389–392, Glasgow, United Kingdom.