TRAINING IRCAM'S SCORE FOLLOWER

Arshia Cont, Diemo Schwarz, Norbert Schnell

Ircam – Centre Pompidou
Real-time Applications Team
1 place Igor Stravinsky, Paris 75004.
{acont,schwarz,schnell}@ircam.fr

ABSTRACT

This paper describes our attempt to make the *Hidden Markov Model (HMM)* score following system developed at Ircam sensible to past experiences in order to obtain better audio to score real-time alignment for musical applications. A new observation modeling based on Gaussian Mixture Models is developed which is trainable using a learning algorithm we would call *automatic discriminative training*. The novelty of this system lies in the fact that this method, unlike classical methods for *HMM* training, is not concerned with modeling the music signal but with correctly choosing the sequence of music events that was performed. Besides obtaining better alignment, new system's parameters are controllable in a physical manner and the training algorithm learns different styles of music performance as discussed.

1. INTRODUCTION

Score following is the real-time alignment of a known musical score to the audio signal produced by a musician playing this score. Score following has been studied for almost 20 years now. The goal is to simulate the behavior of a musician playing with another, a "synthetic performer," to create a virtual accompanist that follows the score of the human musician. For an introduction and state of the art on score following and details of the the system developed by Ircam's Real-Time Applications team, we refer the curious reader to [1] and Chapter 1 in [2]. In this paper, we introduce the learning algorithm used for Ircam's score follower and its outcomes.

We begin this paper by a review of past attempts in score following literature, focusing on the adaptability and learning aspects of the algorithms, specially of importance for our work. This section is followed by an overview of our approach and objective towards training leading to a new *observation modeling* for score following. After reviewing the proposed architecture, we introduce a learning algorithm called *automatic discriminative training* which conforms to the practical criteria of a score following system. The novelty of this system lies in the fact that this method, unlike classical methods for *HMM* training, is not concerned with modeling the music signal but with correctly choosing the sequence of music events that was performed. In this manner, using a *discrimination* process we attempt to model class boundaries rather than constructing an accurate model for each class. Finally, we demonstrate some results and evaluations of the new system.

This work was done in the framework of the project SemanticHIFI, funded by the European Commisson.

2. BACKGROUND

The first debate on learning in the context of score following occurred in Vercoe and Puckette's historical article [3]. Their learning method, interestingly statistical, allows the synthetic performer to rehearse a work with the live performer and thus provide an effective performance, called "post-performance memory messaging." This non real-time program begins by calculating the mean of all onset detections, and subsequently tempo matching the mean-corrected deviations to the original score. The standard deviation of the original onset regularities is then computed and used to weaken the importance of each performed event. When subsequent rehearsal takes place, the system uses these weighted values to influence the computation of its least-square fit for metrical prediction. While in Roger Dannenberg's works before 1997 (or more precisely before the move to a statistical system) there is no report of an explicit training, in Puckette's article [4], there is evidence of off-line parameter control in three instances: defining the weights used on each constant-Q filter associated with a partial of a pitch in the score, the curve-fitting procedure used to obtain a sharper estimate of f_0 and threshold used for the input level of the sung voice. According to the article, he did not envision any learning methods to obtain the mentioned parameters. In the first two instances he uses trial and error to obtain global parameters satisfying desired behavior and the threshold is set by hand during performance. Note that in different performances of the same piece, it is this hand-setting of parameters which correlates to the performance style of the musician.

By moving to the probabilistic or statistical score followers, the concept of training becomes more inherent. In Dannenberg and Grubb's score follower [5], the probability density functions (PDFs) should be obtained in advance and are good candidates for an automatic learning algorithm. In their article, they report three different PDFs in use and they define alternative methods to obtain them, using information based on intuition and experience and information based on empirical investigations of actual performances. A total of 20 recorded performances were used and their pitch detected and *hand-labeled* time alignment is used to provide an observation distribution for actual pitch given a scored pitch and the required *PDF*s would be calculated from these hand-discriminated data.

In the *HMM* score following system of Raphael [6], he trains his statistics (or features in our system's terminology) using a *posterior marginal distribution* $\{p(x_k|\mathbf{y})\}$ to re-estimate his feature probabilities in an iterative manner. In his iterative training he uses *signatures* assigned to each frame for discrimination but no parsing is applied beforehand.

3. APPROACH

Training in the context of score following is to adapt its parameters to a certain style of performance and a certain piece of music. As developed for music performance situations, this system is created to realize the music as opposed to selecting music to demonstrate the technology.

In this respect, we envision a system which adapts itself to correct parameters using a database of sound files of previous performances of the same piece or in the case of a creation, of recorded rehearsals. After the offline and automatic learning, the system is adapted to a certain style of performance, and thus provides better alignment with the score in real-time.

Figure 1 shows a general diagram of Ircam's current score follower as a refinement of the model described in [7]. In our approach to the described problem, we have refined the *observation modeling* (upper block) in order to obtain the desired architecture. The *decision and alignment block* (lower block) is described in detail in [7].

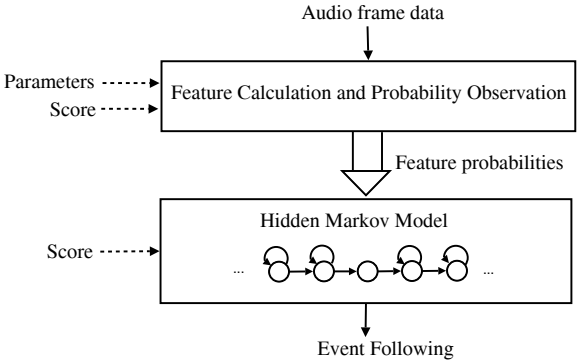

Fig. 1. General diagram of Ircam's score following system

4. OBSERVATION MODELING

Observation in the context of our system consists of calculating features from the audio spectrum in real-time and associate the desired probabilities for low-level HMM states. Low-level states used in our system are *attack*, *sustain* and *rests* for each note in the score; and spectrum features used are *Log of Energy*, *Spectral Balance* and *Peak Structure Match (PSM)*. We will not go into implementation details of the mentioned features which are described in [1, 2], but focus on the learning aspect of the architecture.

The observation process can be seen as a dimension reduction process where a frame of our data, or the FFT points, lies in a high dimensional space, $\Re^J$ where J=2048. In this way, we can consider the features as vector valued functions, mapping the high dimensional space into a much lower dimensional space, or more precisely to $2 + N$ dimensions where N is the number of different notes present in the score for the *PSM* feature. Another way to look at the observation process is to consider it as a probability mapping between the feature values and low-level state probabilities. A diagram of the observation process is demonstrated in Figure 2.

In this model, we calculate the low-level feature probabilities associated with each feature which in terms would be multiplied

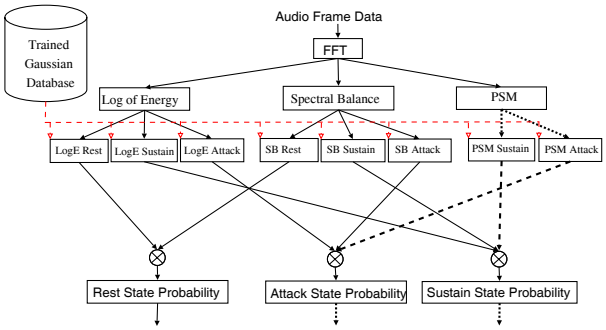

Fig. 2. Probability Observation Diagram

to obtain a certain low-level state probability. As an example, the *Log of Energy* feature will give three probabilities *Log of Energy for Attack*, *Log of Energy for Sustain* and *Log of Energy for Rests*.

In order to calculate probabilities, each low-level state feature probability (third layer in Figure 2) is using probability mapping functions from a database of stored trained parameters. The probability mapping is derived from Gaussians in forms of *cumulative distribution functions (CDFs)*, inverse cumulative distribution functions or PDFs depending on the heuristics associated with each feature state. This architecture is inspired by Gaussian Mixture Models. Note that the dimension of each model used is one at this time.

By this modeling we have assumed that the low-level states' attributes are global which is not totally true and would probably fail in extreme cases. However, due to a probabilistic approach, training the parameters over these cases would solve the problem in most cases we have encountered. Another assumption made is the conditional independence among the features, responsible for the final multiplication of the feature as in Figure 2.

5. TRAINING THE SCORE FOLLOWER

In an ideal training, the system runs on a huge database of *aligned* sound files and adapts its parameters to the performance. In this case, the training is usually supervised and is dependent on system architecture. However, in a musical situation dealing with traditions of music rehearsals and performances, such an ideal procedure would not be possible [2]. In this context, the training will be offline and would use the audio data recorded during rehearsals to train itself.

5.1. The automatic discriminative training

In score following we are not concerned with estimating the joint density of the music data, but are interested in the posterior probability of a musical sequence using the acoustic data. More informally, we are not finally concerned with modeling the music signal, but with correctly choosing the sequence of music events that was performed. Translating this concern to a local level, rather than constructing the set of *PDFs* that best describe the data, we are interested in ensuring that the correct *HMM* state is the most probable (according to the model) for each frame.

This leads us to a *discriminative training* criterion. This criterion has been described in [9] among others. Discriminative training attempts to model the class boundaries — learn the distinction

between classes — rather than construct as accurate a model as possible for each class. In practice this results in an algorithm that minimizes the likelihood of incorrect, competing models as well as maximizing the likelihood of the correct model.

While most discriminative training methods are supervised, for portability and practical issues, it should be automatic if not unsupervised. For this reason, we introduce an automatic supervision over training by constructing a *discrimination knowledge* by an alternative algorithm which forces each model to its boundaries and discriminates feature observations. *Yin* [10] has been chosen as this algorithm to provide discrimination knowledge.

Figure 3 shows a diagram of different steps of this training. The inputs of this training are audio files plus a music score. There are two main cores to this system: *Discrimination* and *Training*.

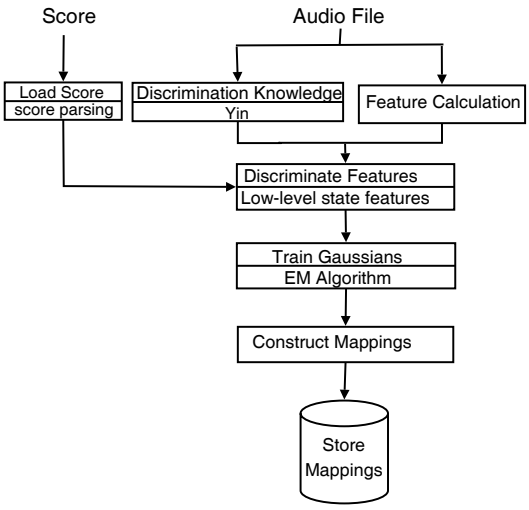

Fig. 3. Automatic Discriminative Training Diagram

5.2. Discrimination

Using discrimination, we aim to distinguish low-level states in the feature domain. In this process, as part of the training, a set of states and their corresponding observations would be obtained without actually segmenting or labeling the performance. The *Yin* algorithm [10] is used as the base knowledge. *Yin* is originally a monophonic fundamental frequency estimator and provides fairly good measures of aperiodicity of each analysis frame. By a one-to-one correspondence between the observed data frames and *Yin*'s analysis frames, and using *Yin*'s information for each frame we decide on the type of the associated low-level state (*Attack*, *sustain* and *release*) for each note in the score.

Note that *discrimination* in this context is implicit and has little to do with the literature on speech discriminative training algorithms. As an analogy, this work is comparable to unsupervised model adaptation algorithms in speech where model parameters are adjusted on the basis of unlabeled training data by making a preliminary recognition. In this way, *discrimination knowledge* refers to unsupervised labeling of the audio file associated with *HMM* low-level states.

Figure 4 shows *Yin*'s f_0 output together with score information as bands for each different note in the score (an extract of *En Echo* by Philippe Manoury). These bands are used to find which notes the frame indices belong to. The aperiodicity measure for each frame discriminates *release* and *note* events and if the detected note meets a minimum time length, about 20 frames, proximity of the first index would be marked as the *attack* frame indices as well as the rest for *sustain* frames. Using these indices, we discriminate *attack*, *release* and *sustain* frames from each feature's observation. Obviously, each observation frame is assumed to have only one state type.

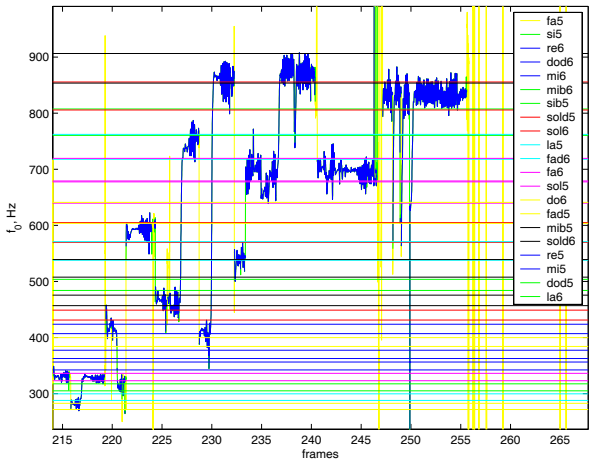

Fig. 4. Discrimination using *Yin*

A final remark on Figure 4 is the noisiness of *Yin*'s measurement which underlies why it is not being used on the first hand in the system itself and the fact that this noisiness will be uncovered during training due to the statistical nature of the algorithm.

5.3. Training

Having all features discriminated, we are ready to train the *Gaussians*. We evade using fitting algorithms due for robustness issues and use *EM Algorithm* [11] to construct the *Gaussians* on observed discriminated features.

The result of the training is a set of *PDF*s that correspond to each low-level state feature. We go further and construct structures containing μ and σ values for each *PDF* as well as the corresponding type of *probability mapping* for each state feature. This data will be stored in a database which will be used in the real-time score follower's *observation* block as shown in Figure 2.

6. EVALUATION AND RESULTS

An evaluation of a score following system is a wide topic. Recently, we discussed the issue of evaluation in an article published at the *NIME* conference [1], where *objective* and *subjective* evaluation was discussed, suggesting a framework for evaluation of different existing systems.

Evaluating the proposed system along with training results would limit the developers to a *subjective* evaluation and comparison to previous alignment results and on critical musical phrases. Figure 5 demonstrates this evaluation for two critical music phrases from Philippe Manoury's piece *En Echo* for Soprano and live electronics for the previous system used for Ircam's score follower de-

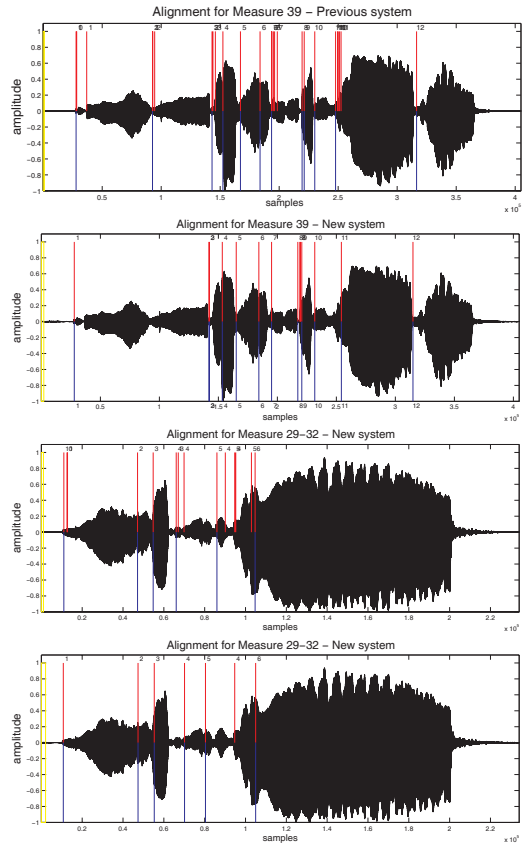

Fig. 5. Alignment results of the previous system and new system, using the proposed observation block and training results, on measure 39 and measures 29-32 of part I of *En Echo*. Vertical lines with numbers demonstrate the segmentations associated with the note number in the score.

scribed in [1] and the system proposed in this article after training. Vertical lines show the segmentation results. In overall, the stability of the system has increased and noisiness has been reduced, essential features for real-time following. Alignment is also improved in general and specially for fast phrases.

One important outcome of this learning algorithm is the ability to model and differentiate different styles of performance of a piece specific to different musicians. For a detailed discussion of this feature, we refer the curious reader to our other article in [12]. This algorithm is independent of the system's architecture.

7. CONCLUSION

In this paper and in the context of a statistical *HMM* score follower developed at Ircam, we present a new approach for the *observation modeling* which can articulate specific behavior of the musician in a controllable manner.

Using this approach, a learning algorithm called *automatic discriminative training* was implemented which conforms to the practical criteria of a score following system. The novelty of this system lies in the fact that this method, unlike classical methods for *HMM* training, is not concerned with modeling the music signal but with correctly choosing the sequence of music events that was performed. The proposed training is independent of system's architecture and has led to improvements in real-time alignment. The system tends to model the margins of different styles of performance to a good extent and moreover, might be a point of departure for further studies in the context of learning algorithms for audio signal processing.

8. ACKNOWLEDGMENTS

We are grateful to Philippe Manoury, Serge Lemouton and Andrew Gerzso without whose valuable comments and presence during test sessions the project could not have advanced. We would like to acknowledge Nicola Orio's ground laying work as the founder of this research project.

9. REFERENCES

[1] Nicola Orio, Serge Lemouton, Diemo Schwarz, and Norbert Schnell, "Score Following: State of the Art and New Developments," in *Proceedings of the International Conference on New Interfaces for Musical Expression (NIME)*, Montreal, Canada, May 2003.

[2] Arshia Cont, "Improvement of observation modeling for score following," M.S. thesis, University of Paris 6, IRCAM, Paris., 2004.

[3] Barry Vercoe and Miller Puckette, "Synthetic Rehearsal: Training the Synthetic Performer," in *Proceedings of the ICMC*, 1985, pp. 275–278.

[4] Miller Puckette, "Score Following Using the Sung Voice," in *Proceedings of the ICMC*, 1995, pp. 199–200.

[5] Lorin Grubb and Roger B. Dannenberg, "A Stochastic Method of Tracking a Vocal Performer," in *Proceedings of the ICMC*, 1997, pp. 301–308.

[6] Christopher Raphael, "Automatic Segmentation of Acoustic Musical Signals Using Hidden Markov Models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 4, pp. 360–370, 1999.

[7] Nicola Orio and F. Déchelle, "Score Following Using Spectral Analysis and Hidden Markov Models," in *Proceedings of the ICMC*, Havana, Cuba, 2001.

[8] Nicola Orio and Diemo Schwarz, "Alignment of Monophonic and Polypophonic Music to a Score," in *Proceedings of the ICMC*, Havana, Cuba, 2001.

[9] S. Renals, N. Morgan, H. Bourlard, M. Cohen, and H. Franco, "Connectionist probability estimators in HMM speech recognition," *IEEE Transactions Speech and Audio Processing*, 1993.

[10] Alain de Cheveigne and H. Kawahara, "YIN, a fundamental frequency estimator for speech and music," *J. Acoust. Soc. Am.*, vol. 111, pp. 1917–1930, 2002.

[11] A.P. Dempster, N. M. Laird, and D. B. Rubin, "maximum likelihood from incomplete data via the EM algorithm," *Journal of Royal Statistical Society*, vol. 39, no. B, pp. 1–38, 1977.

[12] Arshia Cont, Diemo Schwarz, and Norbert Schnell, "Training IRCAM's Score Follower," in *AAAI Fall Symposium on Style and Meaning in Art, Language and Music*, October 2004.