IMPROVING MIDI-AUDIO ALIGNMENT WITH ACOUSTIC FEATURES

Johanna Devaney[*]

DDMAL, CIRMMT
Schulich School of Music,
McGill University, Montreal, QC, Canada
devaney@music.mcgill.ca

Michael I. Mandel and Daniel P.W. Ellis

LabROSA
Department of Electrical Engineering
Columbia University, New York, NY, USA
{mim, dpwe}@ee.columbia.edu

ABSTRACT

This paper describes a technique to improve the accuracy of dynamic time warping-based MIDI-audio alignment. The technique implements a hidden Markov model that uses aperiodicity and power estimates from the signal as observations and the results of a dynamic time warping alignment as a prior. In addition to improving the overall alignment, this technique also identifies the transient and steady state sections of the note. This information is important for describing various aspects of a musical performance, including both pitch and rhythm.

Index Terms— MIDI-audio alignment, music performance, singing, dynamic time warping, HMMs, frequency estimation.

1. INTRODUCTION

Precise descriptions of music signals are essential for studying both acoustical and interpretative aspects of musical performance [1]. MIDI-audio alignment is a useful tool for identifying note boundaries, i.e., onsets and offsets [2]. This information can be used directly for timing-based studies and is also important for pitch-based research in the absence of robust transcription methods [3]. However, the current state of the art for MIDI-audio alignment is not yet accurate enough to provide reliable data for such studies. The options for identifying note onsets and offsets are either laborious manual analysis or a quasi-automated process, where the results of an alignment system are corrected by hand. This paper describes a technique for improving the accuracy of MIDI-audio alignment by using known acoustical properties of the signal to train an HMM to identify silence, transient, and steady state portions of each note. The implementation in this paper is for solo singing voice, though the technique could be applied to other instruments by modifying the acoustical features and to polyphonic signals with the use of an algorithm capable of producing the required acoustical descriptions.

2. EARLIER WORK

Work in the area of MIDI-audio alignment can be divided into two distinct approaches. Online approaches, or score followers, are typically applied in live performance contexts [4] while offline approaches are used for a range of applications by the music information retrieval community, including audio database searches [5] and digital libraries [6]. Typically graphical models, including hidden Markov models (HMMs), have been used for online applications [7-10] whereas the related, more constrained, technique of dynamic time warping (DTW) has predominantly been used for offline techniques [5,6,11,12]. The online approaches often sacrifice precision for efficiency, low latency and robustness in the face of incorrect notes [13,14]. The offline approaches are more precise, but do not work particularly well for non-percussive instruments, such as the singing voice [15] or stringed instruments.

This approach of using an initial alignment to guide a secondary process is similar in this respect to the bootstrapping algorithm for onset detection described in [16], where an initial DTW alignment is used to establish note boundaries that are in turn used to train a multi-layer perceptron neural network for onset detection. Similarly, HMMs have previously been used for describing signals containing the voice in [17] and [18]. In [17], a three-state HMM was implemented to model the phonemes of hummed notes for a query-by-humming application. [18] deals explicitly with transcription of the singing voice and uses a three state note event HMM and a four component rest event GMM trained on examples of singing and no-singing audio frames respectively.

3. IMPROVING ALIGNMENT ACCURACY

The goal of this work is to improve the accuracy of an initial DTW alignment with a Hidden Markov Model that models the acoustical properties of the singing voice. Since the HMM performs only local adjustments to the alignment, a relatively accurate initial alignment is important for this technique and achievable with DTW. Through a comparative evaluation of different features, we determined that the use of peak spectral difference [8] for the alignment feature produced the more accurate DTW alignment for the singing voice. The HMM was implemented in Matlab with Kevin Murphy's HMM Toolbox [19] using periodicity and power estimates from Alain deCheveigne's YIN algorithm [20].

[*] This work was supported by the Center for Research in Music Media and Technology (CIRMMT), the Fonds de recherche sur la société et la culture (FQRSC), the Social Sciences and Humanities Research Council of Canada (SSHRC), and the National Science Foundation under grant IIS-0713334. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the sponsors.

3.1. Acoustical properties of the singing voice

The design of the HMM was based on the acoustical properties of the singing voice. As a result, this implementation is optimized for the singing voice and would require some adjustment to work with other instruments. The amplitude envelope and periodic characteristics of a sung note are influenced by the words that are being sung. The four acoustic events modeled for this system (silence, breath, transient, sustain/steady state) are shown in Figure 1. Transients occur when a consonant starts or ends a syllable, while vowels produce the steady state portion of the note. The type of consonant, voiced or unvoiced, affects the characteristics of the transient, as does the particular manner in which the singer attacks or enunciates the consonant. The motivation for identifying transients is to determine where the voiced section of the note begins for estimating a single fundamental frequency of the duration of the note. In order to facilitate this, only unvoiced consonants have been modeled as transients in this system.

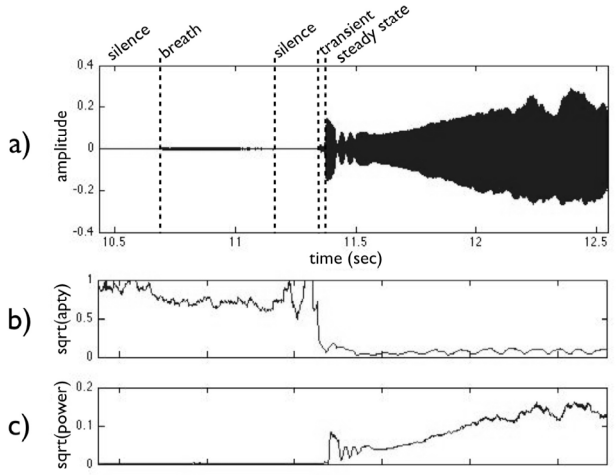

Figure 1:Time domain representation of a sung note's waveform (a), aperiodicity (b), and power (c), with the HMM states labeled.

3.2. States

The basic implementation of this HMM has three states: silence, transient, steady state, see Figure 2. An optional fourth state, breath, was introduced experimentally which improved results in some cases, but should be considered an optimization, rather than an essential component of the model, see Figure 3. A second silence after the breath state is added in this state sequence to reflect the common practice among singers of briefly holding the inhaled breath before singing the next note.

The transition probability values were calculated from a superset of the music used in the experiments in section 5, including Schubert's Ave Maria and a Latin mass by Machaut. The silence, breath, transient, and steady state portions of these pieces were hand-labeled by the researchers. Self-loop probabilities, the probability that a state with repeat instead of changing, were estimated from the average duration of each state in 90 seconds of audio. Non self-loop probabilities were estimated from summary statistics of 318 notes from these scores. Specifically, the transition probabilities to the transient states were set to reflect the likelihood of syllables beginning and ending with consonants in the Latin text. And transition probabilities to the silences were based on the average frequency of rests in the score, as it was assumed that in the legato singing style that dominates the singing voice literature, silences would only occur at rest or breath marks.

Two versions of the state sequences were implemented. The first algorithm, see section 5.1, allows each state to be visited for each note. The second algorithm, section 5.2, was determined by the particular lyrics being sung, transients were only inserted when a consonant began or ended a syllable and silences (and breaths for experiment 2b) were inserted only at the end of phrases. The state sequence for the opening phrase of Schubert's Ave Maria is shown in Figure 4.

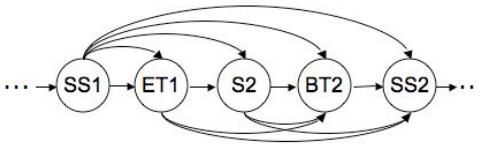

Figure 2: Three-state basic state sequence seed: steady state (SS), transient (T), silence (S). Both ending transient (ET) and the starting transient (ST) have the same observation distribution.

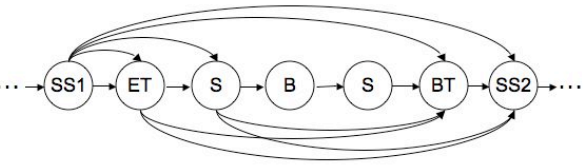

Figure 3: Basic state sequence seed plus breath (B).

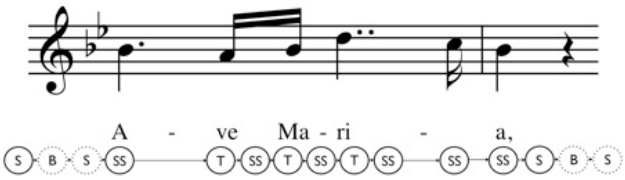

Figure 4: State sequence adapted to sung text.

3.3. Observations

The observations for the HMM are the square root of periodicity and power estimates provided by the YIN algorithm for each frame. YIN estimates fundamental frequency by measuring the self-similarity of a signal over time. While standard autocorrelation uses an inner product to measure similarity, YIN uses the squared difference to measure dissimilarity.

YIN was run on audio sampled at 44,100 with a frame size of 10ms and a hop size of 0.7ms. The mean and covariance values for each frame were calculated by isolating representative examples of silence, transient, steady state, and breath from recordings by different singers. In total, 2.25s of data were used to calculate the means and variances for silence, 13.4s for steady state, 0.47s for transients, and 3.83s for breath.

3.4. Prior

The initial DTW alignment is used as a prior to guide the HMM (see Figure 4). DTW warps two sequences to match each other while minimizing the number of insertions and deletions necessary to align the sequences. The use of the DTW alignment obviates the need to encode information about the score in the HMM. By assuming that the DTW alignment is roughly correct, it is not necessary to encode pitch specific information into the HMM. This drastically simplifies the problem that the HMM has to address, and thus simplifies the design of the HMM as it allows the same HMM seed to be used for each note. One issue with this approach is that it cannot adjust the initial alignment by more than one note, so the initial alignment has to be relatively accurate.

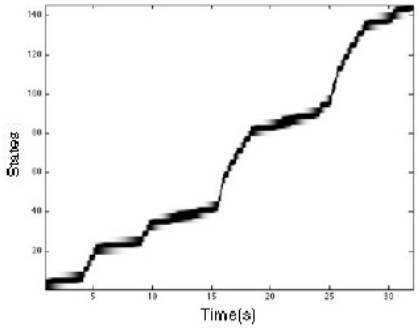

Figure 4: Visualization of the DTW alignment implemented as a prior for the HMM

The prior is created by placing a rectangular window with half a Gaussian on each side over the note positions estimated by the DTW alignment. Each state has a different set of rules governing the placement and width of the windows and half Gaussians, as detailed in Table 1.

	5% start	100% start	100% end	5% end
Silence (and Breath)	50% btwn N-1 On and N-1 Off	N-1 Off	N On	50% btwn N On and N Off
Opening Transient	N-1 Off	75% btwn N-1 Off and N On	25% btwn N On and N Off	N Off
Steady State	N-1 Off	N On	N Off	N+1 On
Closing Transient	N On	75% btwn N On and N Off	25% btwn N Off and N+1 On	N+1 On

Table 1: Gaussian distributions for the creation of a prior from the DTW alignment. N is the current note number.

4. EVALUATION

4.1. Test Data

Three annotated recordings of the opening of Schubert's Ave Maria by three different singers were used to evaluate the system. The annotations were done manually using Audacity [20]. All of the singers had soprano voices, one was a professional and the other two were undergraduate vocal majors. The singers' exhibited differences in overall timbre, attack time (transient length), and vibrato rates.

4.2. Algorithm One – General state sequence

In this first algorithm, each note is modeled with a complete set of states. This is the baseline test, to evaluate whether performance is improved when the text is taken into account (algorithm two). The first version of this algorithm (1a) uses the basic three-state HMM model, as per Figure 2, and the second (1b) adds the optional breath state, as per Figure 3.

4.3. Algorithm Two – State sequence adapted to sung text

In the second algorithm, the state space is modified based on the presence of consonants in the sung text and phrase endings or rests in the score, see Figure 4. As with algorithm one, this algorithm was run both with the basic three-state HMM (2a) and with the optional breath state added (2b).

4.4. Results

The results of the experiments are detailed in Table 2, which provides the 2.5, 25, 50, 75, and 97.5 percentiles of the absolute difference between the manually annotated ground truth for both the experiments and the original DTW alignment.

Percentile	2.5	25	50	75	97.5
DTW	3.2	32.6	52.3	87.9	478.7
1a General w/o breath	1.6	13.1	41.8	88.8	564.1
1b General w/breath	1.9	13.7	47.4	117.8	923.8
2a Textual w/o breath	1.6	13.1	27.8	78.0	506.0
2b Textual w/breath	2.1	13.7	41.8	91.3	923.1

Table 2: Results from Algorithms 1 and 2 compared to the original dynamic time warping alignment in milliseconds.

In general, both algorithms provided greater alignment accuracy than the initial DTW alignment, although the outliers (around the 97.5th percentile) for the unmodified state sequence of algorithm one were less accurate than the DTW. There was also a consistent improvement in performance by the modified state sequence used in algorithm two over the unmodified sequence in algorithm one. This was largely to be expected, as in the first algorithm the HMM had the freedom to select a state that would not have occurred at certain points in the recorded performance. The addition of the breath state did not increase the accuracy of the alignment, in fact it led to a small number of quite severe misalignments. On inspection, these misalignments occurred at the silence-breath-silence states and not in the transient and steady state portions of the notes that we are concerned with.

A visual demonstration of the improvement in alignment can be seen in Figure 6. Here the boxes indicate the DTW alignment, the red horizontal lines are the silences predicted by the model, the green lines are the transients, and the blue lines are the steady state portion of the notes. At approx. 400ms, 800ms, and 1500ms (labels 1, 2, and 3 respectively) the DTW alignment estimates the offsets too early and the onsets too late and at approx. 1800ms (label 4) the DTW estimates the offset too late. All of these misalignments are corrected by the HMM, also at 1 and 3 the HMM successfully identifies the presence of the transients at the start of the notes.

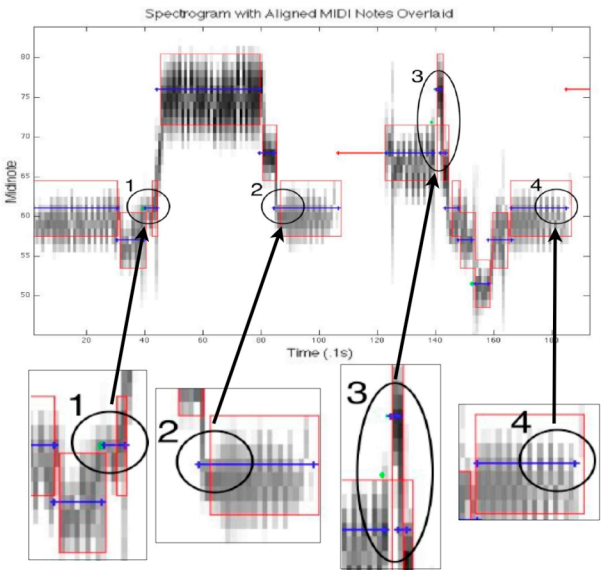

Figure 5: Visualization of the performance of the Algorithm 2a versus the DTW alignment.

5. DISCUSSION

Closer examination of where the HMM made incorrect state identifications revealed that some voiced consonants introduced a considerable amount of 'noise' in steady state sections, i.e., when the consonants are rolled. The implementation of the transient state predominantly modeled unvoiced consonants, while allowing for the possibility of several frames of noise at the start of a voiced consonant. The implementation of the steady state portion covered both voiced consonants and vowels. The reason for this is that the voiced consonants contribute to the perceived pitch. There is also some ambiguity present in the ground truth. Onsets and offsets in the singing voice are notoriously difficult to identify [22], which may affect the accuracy of the ground truth, and thus the results of the experiments, by several tens of milliseconds.

6. CONCLUSIONS AND FUTURE WORK

Overall, the 3-state HMM algorithm was able to improve the results of the standard DTW alignment, decreasing the median alignment error from 52 to 42 ms. When a simple model of the phonetics of the lyrics was taken into consideration, the median error was further reduced to 28 ms. This is promising and generally sufficient for determining start and end points of steady state portions of notes when calculating the perceived fundamental frequency. Plans for future work include experimenting with extensions to the HMM to differentiate explicitly between voiced and unvoiced consonances. This information will be useful when calculating a single perceived fundamental frequency over the duration of a note.

7. REFERENCES

[1] Seashore, C. 1936. *Objective analysis of musical performance*. New York, NY: McGraw–Hill.

[2] Scheirer, E.D. 1998. Using musical knowledge to extract expressive performance information from audio recordings. In H. Okuno and D. Rosenthal (eds.) *Readings in Computational Auditory Scene Analysis*. Mahwah, NJ: Lawrence Erlbaum.

[3] Devaney, J. and D.P.W. Ellis. 2008. An empirical approach to studying intonation tendencies in polyphonic vocal performances. *Journal of the Interdisciplinary Music Studies*. 141–56.

[4] Dannenberg, R. and C. Raphael. 2006. Music score alignment and computer accompaniment. *Communications of the ACM*. 49(8): 38–43.

[5] Hu, N., R. Dannenberg, and G. Tzanetakis. 2003. Polyphonic audio matching and alignment for music retrieval. In *Proc. WASPAA*. 185–8.

[6] Kurth, F., M. Müller, D. Damm, C. Fremerey, A. Ribbrock, and M. Clausen. 2004. Syncplayer – An advanced system for multimodal music access. In *Proc. ISMIR*. 381–8.

[7] Cano, P., A. Loscos, and J. Bonada. 1999. Score-performance matching using HMMs. In *Proc. ICMC*. 441–4.

[8] Orio, N., and F. Déchelle. 2001. Score following using spectral analysis and hidden Markov models. In *Proc. ICMC*. 151–4.

[9] Raphael, C. 2004. A hybrid graphical model for aligning polyphonic audio with musical scores. In *Proc. ISMIR*. 387–94.

[10] Peeling, P., T. Cemgil, and S. Godsill. 2007. A probabilistic framework for matching music representations. In *Proc. ISMIR*. 267–72.

[11] Orio, N., and D. Schwarz. 2001. Alignment of monophonic and polyphonic music to a score. In *Proc. ICMC*. 155–8.

[12] Turetsky, R., and D.P.W. Ellis. 2003. Ground-truth transcriptions of real music from force-aligned MIDI synthesis. In *Proc. ISMIR*. 135–42.

[13] Cont. A, D. Schwarz, N. Schnell, and C. Raphael. 2007. Evaluation of real-time audio-to-score alignment. In *Proc. ISMIR*. 315–6.

[14] Downie, J. 2008. MIREX 2008: Real-time audio to score alignment (a.k.a. score following). http://www.music-ir.org/mirex/2008/index.php/Realtime_Audio_to_Score_Alignment_(a.k.a_Score_Following) (accessed April 13, 2009).

[15] Devaney, J., and D.P.W. Ellis. 2009. Handling asynchrony in audio-score alignment. In *Proc. ICMC*. To appear.

[16] Hu, N. and R. Dannenberg. 2006. Bootstrap leaning for accurate onset detection. *Machine Learning*. 65: 457–71.

[17] H. Shih, S.S. Narayanan, and C.-C. J. Kuo. 2003. A statistical multidimensional humming transcription using phone level hidden Markov models for query by humming systems. In *Proc. IEEE International Conference on Multimedia*. 1: 61–4.

[18] Ryynänen, M.P. and A.P. Klapuri. 2006. Transcription of the Singing Melody in Polyphonic Music. In *Proc. ISMIR*. 222–7.

[19] http://www.cs.ubc.ca/~murphyk/Software/HMM/hmm.html

[20] de Cheveigné, A. and H. Kawahara. 2002. YIN, a fundamental frequency estimator for speech and music. *Journal of the Society of the Acoustical Society of America*. 111(4): 1917–30.

[21] http://audacity.sourceforge.net/

[22] Toh, C.C., B. Zhang, Y. Wang. 2008. Multiple-feature fusion based on onset detection for solo singing voice. In *Proc. ISMIR*. 515–20.