# A PROBABILISTIC FRAMEWORK FOR MATCHING MUSIC REPRESENTATIONS

**Paul Peeling** **A. Taylan Cemgil** **Simon Godsill**
Signal Processing and Communications Laboratory, Department of Engineering,
Cambridge University, Trumpington Street, Cambridge CB2 1PZ, United Kingdom
{php23,atc27,sjg}@eng.cam.ac.uk

**ABSTRACT**

In this paper we introduce a probabilistic framework for matching different music representations (score, MIDI, audio) by incorporating models of how one musical representation might be rendered from another. We propose a dynamical hidden Markov model for the score pointer as a prior, and two observation models, the first based on matching spectrogram data to a trained template, the second detecting damped sinusoids within a frame of audio by subspace methods. The resulting Bayesian framework is robust to local variations in tempo, and can be used for a wide variety of applications. We evaluate both methods in a score alignment context by inferring the posterior distribution of the current position in the score exactly. The spectrogram method is shown to infer the score position reliably with minimal computation, and the damped sinusoid model is able to pinpoint the positions of score events in the audio with a high level of timing accuracy.

## 1 INTRODUCTION

Musical information is roughly represented in one of three ways: a score, which is a symbolic representation, a MIDI file, which represents discrete musical events with more precise timing information, and sampled audio, which is the most faithful representation of the sound produced. There are many applications for which we would like to match a number of pieces of music with different representations together. For example, score alignment [14, 10, 9] is the matching of a score representation to the audio representation of the same music. Often in practice, this problem can be reformulated as matching a MIDI representation to audio, assuming the MIDI is quantized to discrete positions and accurately represents the score.

In all these applications, the underlying factor which is responsible for causing mismatches between different representations is an unknown tempo process. For example, in the score alignment problem, the tempo of a MIDI representation evolves independently from that of the audio, hence dynamic time warping (DTW) strategies have been popular [19, 4, 11]. Audio synchronization, where two audio representations with different tempi are matched, can also be treated by these strategies [15].

Dynamic time warping (DTW) schemes rely on minimizing an explicit matching function by dynamic programming and may encounter difficulties when unexpected events occur, which are not captured in the matching criteria, for example, mistakes made by a player when performing a piece from a score, repeats made in a concert but not during rehearsals, improvisation sections, pauses and reruns, and so on. A complete probabilistic model for music representation enables inclusion of such types of events as *a priori* information and facilitates learning from data, hence potentially a more robust matching performance can be obtained. Moreover, modern and powerful inference techniques can be developed in cases where the model size becomes large so as not to admit exact computation.

In this paper we introduce a probabilistic framework which will allow us to match different music representations in a Bayesian setting. We begin by considering the fundamental representation of music as the score, and construct a prior model of how this representation evolves in time during a performance. One such approach has been developed by Raphael [18], where a probabilistic dynamical model is applied to the tempo of the audio, with the expected timing of events based on the score in a score alignment context. Here we consider the evolution of the position of a 'score position pointer' through time, adopting an approach similar to that of [8, 2, 20]. We define the 'score pointer' as an unobserved random variable over score positions evolving according to unknown velocities (tempi). This model differs from previous approaches in the way the tempi are represented. In Section 2 we describe this dynamical model for the score pointer, formulated as a hidden Markov model [17]. Given the formulation it is conceptually straightforward to develop online, offline and fixed-lag applications using standard exact or approximate inference methodology.

In the Bayesian setting, we also require an observation model which assigns a likelihood value to observed data from a different music representation given the current state of the score position pointer. In Section 3 we present two such probabilistic models for the generation of music audio from a score, or where practically more appropriate, MIDI. Our approach in this paper will be to constrain our models to allow for exact inference of the posterior dis-

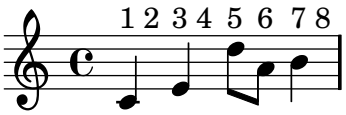


**Figure 1**. State space of the score position pointer $r_k$.

tributions, algorithms for which are provided in Section 4. In future work we will relax these constraints for more elaborate and realistic models, thus requiring approximate inference techniques. such as sequential Monte Carlo [5] or variational Bayes [12]. In Section 5 we demonstrate how to apply the Bayesian models to the score alignment problem, and compare the two observation models on real polyphonic piano audio extracts.

## 2 SCORE POINTER DYNAMICS

We define the score pointer $r_k \in [1, 2, \ldots, R]$ as the position in a musical score at time $k$, measured as the number of eighth notes, sixteenth notes etc. from the beginning of the score. For example, in the simple score in Figure.1 we have $R = 8$ and the unit is an eighth note, the finest score resolution. We represent tempo $t_k \in T$ implicitly by the probability $\pi(t_k)$ that the score pointer $r_k$ moves to the next state. Roughly, when the tempo is fast (slow) the probability to move to the next position is higher (lower). This leads to the following simple dynamics:

$$p(r_k|r_{k-1}, t_k) = \begin{cases} \pi(t_k) & \text{if } r_k = n+1, r_{k-1} = n \\ 1 - \pi(t_k) & \text{if } r_k = n, r_{k-1} = n \\ 0 & \text{otherwise} \end{cases}$$
$$p(r_1|t_1) = 1/R \,\forall\, n \qquad (1)$$

where $R$ is the overall length of the score. The uniform prior (1) allows the performance to begin at any position in the score, which is useful for practical applications. Clearly, one can allow for more elaborate score transition structures such as repeats, improvisation sections and mistakes.

The number of frames for a score pointer transition to take place is a random variable with a geometric distribution with probability of success $\pi(t_k)$. The variance of this distribution is $(1 - \pi(t_k))/\pi(t_k)^2$ which in practice is large enough to account for substantial deviations from predicted score transitions, including the performance halting for some period of time. Hence we only need to consider a small selection of coarse discrete set of tempo values such as $t_k \in \{$ 'fast', 'medium', 'slow' $\}$ to account for a wide range of performance conditions.

## 3 FREQUENCY DOMAIN OBSERVATION MODELS OF AUDIO

In this section, we describe two models to extract frequency-domain features from audio frames.

### 3.1 Generative Spectrogram Model

Here we introduce a model for generating two-element vectors corresponding to the real and imaginary parts of the complex values $s_\nu$ returned by the discrete Fourier transform (DFT) in frequency bins $\nu = 1, \ldots, W$, based on the current score position $r_k$ and a scaling parameter $\lambda_k$ which represents the overall energy in the signal at time $k$. See [13] for an existing approach to a Bayesian model of the spectrogram. In the sequel, we omit the time index for simplicity, please refer to the graphical model in Figure 3. We do this via latent scale parameters $v_\nu$ which describe the energy in each frequency bin, as follows [1]

$$p(v_\nu|r, \lambda) = \mathcal{IG}(v_\nu; a/2, 2/(\lambda\sigma_\nu(r)a)) \qquad (2)$$
$$p(s_\nu|v_\nu) = \mathcal{N}(s_\nu; 0, v_\nu I) \qquad (3)$$

The gain parameter $\lambda$ scales a spectrogram template $\sigma_\nu$ to match $s_\nu$, from which the scale parameters $v_\nu$ are drawn according to (2) with 'tightness' $a$. The spectrogram templates have unit energy, i.e.

$$\sum_{\nu=1}^{W} \sigma_\nu = 1 \qquad (4)$$

See Figure 2 for an example, where high energy regions correspond to the fundamental and partials of the note being played. The spectrogram values are then drawn from a bivariate Gaussian (3) with covariance $v_\nu I$ which is invariant to phase.

One simple method of linking frames of audio together is shown in Figure 3. Realistically the energy in frame $k$ will increase if there is a note onset in the score transition, hence we could add further dependencies $\lambda_k \sim p(\lambda_k|\lambda_{k-1}, r_k, r_{k-1})$, resulting in a changepoint model. See [3] for a example of inference on changepoint models in a music transcription setting. Further links could also be added between the parameters $v_\nu$ in Figure 3 to reflect that the energy in a frequency bin $v_{\nu,k}$ depends on $v_{\nu,k-1}$ in the previous frame and some damping coefficient $\rho$. For this paper however, Figure 3 represents a model for which it will be possible to exactly infer the performance variables $r_k, t_k$ provided we have observed the energy $\lambda_k$ for each frame $k = 1, \ldots, K$. We can estimate $\lambda_k$ from the total energy in the frequency bins for frame $k$, i.e.

[1] Definitions of probability distributions used in this paper

$$\mathcal{IG}(x; \alpha, \beta) = \frac{(1/\beta)^\alpha}{\Gamma(\alpha)} x^{-\alpha-1} \exp(-\frac{1}{\beta x})$$
$$\mathcal{N}(x; \mu, \Sigma) = \frac{1}{|2\pi\Sigma|^{1/2}} \exp\left(-\frac{1}{2}(x-\mu)^T\Sigma^{-1}(x-\mu)\right)$$

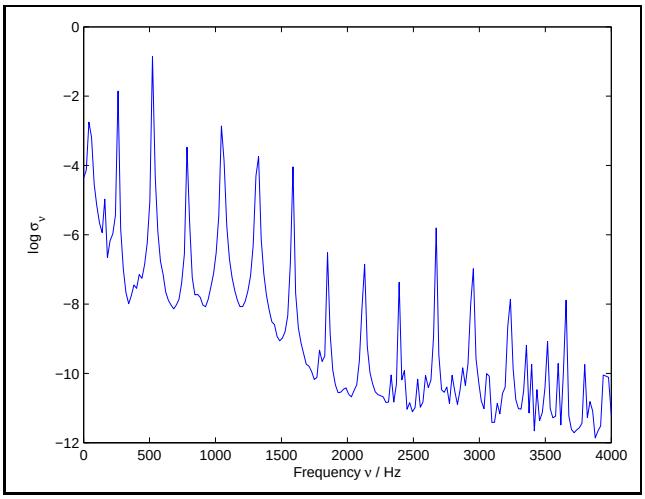


**Figure 2**. Spectrogram template $\sigma_\nu$ of a piano playing middle C (261.6 Hertz) with a sampling frequency of 8000Hz and a framelength of 400 samples (50ms)

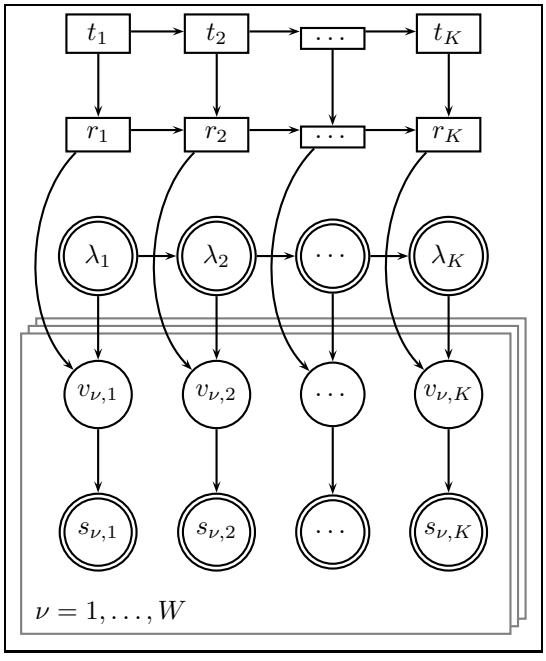


**Figure 3**. Generative spectrogram model. A directed arc between nodes denotes that the second variable is conditionally dependent on the first.

$$\lambda_k = \sum_{\nu=1}^{W} s_{\nu,k}^T s_{\nu,k} \tag{5}$$

As this is only an estimate, it would be possible to treat $\lambda_k$ as an unobserved variable, and use the spectrogram energy (5) as a proposal in an approximate inference scheme. Here, as we will be performing exact inference, it will also be useful to integrate out the latent scale parameters $\nu_k$ in (2) and (3) resulting in a form of Student's $t$-distribution

$$\begin{aligned} p(s_\nu|r,\lambda) &= \int p(s_\nu|v_\nu)p(v_\nu|r,\lambda)dv_\nu \\ &= \frac{\Gamma((a+D)/2)}{(\pi a \lambda\sigma_\nu(r))^{D/2}\Gamma(a/2)}\left(1+\frac{1}{a}\frac{s_\nu^T s_\nu}{\lambda\sigma_\nu(r)}\right)^{-\frac{(a+D)}{2}} \\ &= \mathcal{T}_a(s_\nu; 0, \lambda\sigma_\nu(r)) \end{aligned}$$

In practice, we can train the spectrogram template for a note by taking the spectrogram of a training sample, normalizing each frame so that (4) holds, and computing the mean value in each bin across frames.

### 3.2 Damped Sinusoidal Model

Subspace methods [1] allow high frequency resolution estimates of damped sinusoids present in a signal, by fitting a parameterized model of the form

$$x(t) = \sum_{m=1}^{M} a_m e^{-\rho_m t}\cos(2\pi\omega_m + \phi_m)$$

Hence for a frame of audio, we obtain estimates of the frequencies $\omega_m$, amplitudes $a_m$, phases $\phi_m$ and damping coefficients $\rho_m$ for a model of $M$ sinusoids.

In this paper we will define a model based on the observed frequency values $\omega_m$ and amplitudes $\alpha_m$. Damping coefficients $\rho_m$ could potentially be used to model instrument timbre, but will not be considered here. Given the score position $r$ for the frame, we assume that the frequency and amplitude values are drawn independently from a distribution $p(\omega_m, \alpha_m|r)$, i.e.

$$p(\omega_{1:M}, \alpha_{1:M}|r) = \prod_{m=1}^{M} p(\omega_m, \alpha_m|r)$$

Figure 4 shows the frequency and amplitude data from the ESPRIT subspace method, together with a Gaussian mixture model (GMM) trained on the data. The parameters of the GMM are determined by maximum a posteriori (MAP) estimation, with priors placed on the covariance of each Gaussian component so that the harmonic structure of the subspace data is captured. A uniform 'clutter' component is included to account for spurious detections. To account for different note volumes, the amplitude data

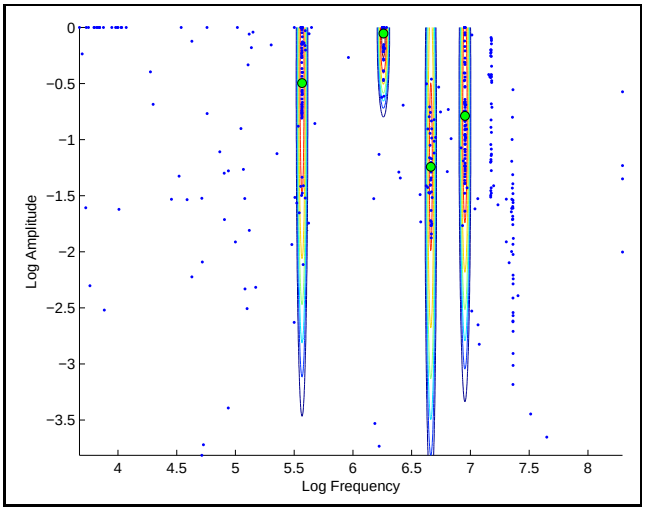


**Figure 4**. Subspace frequency-amplitude data of a piano playing middle C with a sampling frequency of 8000Hz, a framelength of 200 samples and $M = 4$ sinusoids. The Gaussian mixture model trained on this data is indicated by the positions of the means and contours of equal probability. The covariance priors are diagonal and Gaussian: $\sigma_f \sim \mathcal{N}(\sigma_f; 10^{-2}, 10^{-4})$ is the log frequency prior, $\sigma_\alpha \sim \mathcal{N}(\sigma_\alpha; 10^{0}, 10^{-1})$ is the log amplitude prior.

in each frame is scaled so that the maximum sinusoid log amplitude detected in a frame is zero.

A similar model, where the number of sinusoids detected is Poisson rather than fixed at $M$ has been applied to polyphonic music transcription in [16], and is a mathematically sound method for avoiding data association between partials of a musical note and the detected sinusoids.

## 4 INFERENCE

In our Bayesian framework, matching consists of inferring the unknown score position $r_k$ at time $k$, integrating out the tempo value $t_k$. Typically we may wish to determine the most likely score position at time $k$, given past observations $p(r_k|y_{1:k})$, which is known as filtering and can be carried out recursively online, or including all future observations $p(r_k|y_{1:K})$, which is known as smoothing and must be carried out offline, or including some recent observations $p(r_k|y_{1:k+N})$, which is known as fixed-lag smoothing and is practical if a certain amount of latency in the inference is acceptable. We may also wish to predict future values of the score position $p(r_{k+N}|y_{1:k})$ or infer the most likely progression of score positions $p(r_{1:K}|y_{1:K})$ which is known as the *Viterbi* path and is most suitable to offline matching. The computations required for all these related but distinct queries can be viewed in terms of message passing algorithms, and will be described in this section.

The observations are $y_k = \{s_{1:W,k}, \lambda_k\}$ for the spectrogram model, and $y_k = \{\omega_{1:M}, \alpha_{1:M}\}$ for the damped sinusoid model.

By Bayes' theorem, the posterior distribution over the unknown variables $\mathcal{H}_{1:K} \equiv \{r_{1:K}, t_{1:K}\}$ is given by

$$p(\mathcal{H}_{1:K}|y_{1:K}) = \frac{p(y_{1:K}|\mathcal{H}_{1:K})p(\mathcal{H}_{1:K})}{p(y_{1:K})}$$

The marginal filtering density $p(\mathcal{H}_k|y_{1:k})$ can be computed by passing $\alpha_{k|k} \equiv p(\mathcal{H}_k|y_{1:k})p(y_{1:k})$ 'alpha' messages between neighbouring frames:

$$\begin{aligned}
\alpha_{0|0} &= p(\mathcal{H}_0) \\
\alpha_{k|k-1} &= \sum_{\mathcal{H}_{k-1}} p(\mathcal{H}_k|\mathcal{H}_{k-1})\alpha_{k-1|k-1} \\
\alpha_{k|k} &= p(y_k|\mathcal{H}_k)\alpha_{k|k-1}
\end{aligned}$$

We then obtain the desired density up to a normalizing constant by integrating over tempo values $t_k \in T$

$$p(r_k|y_{1:k}) = \sum_{t_k \in T} p(r_k, t_k|y_{1:k}) \propto \sum_{t_k \in T} \alpha_{k|k}$$

The marginal smoothing density $p(\mathcal{H}_k|y_{1:K})$ is computed offline by passing $\beta_{k|k} \equiv p(y_{k+1:K}|\mathcal{H}_k)$ 'beta' messages as follows:

$$\begin{aligned}
\beta_{K|K+1} &= 1 \\
\beta_{k|k} &= p(y_k|\mathcal{H}_k)\beta_{k|k+1} \\
\beta_{k-1|k} &= \sum_{\mathcal{H}_k} p(\mathcal{H}_k|\mathcal{H}_{k-1})\beta_{k|k} \\
p(\mathcal{H}_k|y_{1:K}) &\propto \alpha_{k|k}\beta_{k|k+1}
\end{aligned}$$

The Viterbi path is computed in an analogous manner, where messages from neighbouring frames and observations are combined by taking the maximum rather than summing, i.e.:

$$\begin{aligned}
\alpha_{k|k-1} &= \max_{\mathcal{H}_{k-1}} p(\mathcal{H}_k|\mathcal{H}_{k-1})\alpha_{k-1|k-1} \\
\beta_{k-1|k} &= \max_{\mathcal{H}_k} p(\mathcal{H}_k|\mathcal{H}_{k-1})\beta_{k|k} \\
\operatorname*{argmax}_{\mathcal{H}_{1:K}} p(\mathcal{H}_{1:K}|y_{1:K}) &= \operatorname*{argmax}_{k=1:K} \alpha_{k|k}\beta_{k|k+1}
\end{aligned}$$

## 5 APPLICATIONS

The matching framework introduced in this paper is able to address a wide range of known applications in music information retrieval. We have chosen here to demonstrate score alignment using the observation models discussed above. The aim of score alignment is to infer the score position in an audio extract. For purposes of evaluation we sample an accurate MIDI transcription of the score over a fine set of times, and train our observation models based on the pitch content of the MIDI. Our training data was obtained separately from test data by summing piano audio samples (RWC-MDB-I-2001 No. 01) for each pitch,

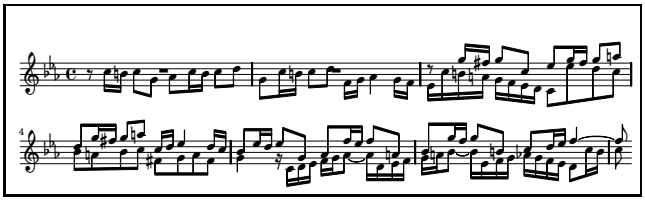

**Figure 5**. Score of an extract from Bach's Well-Tempered Klavier: Book 1 Fugue 2 in C minor. Audio source: Daniel-Ben Pienaar

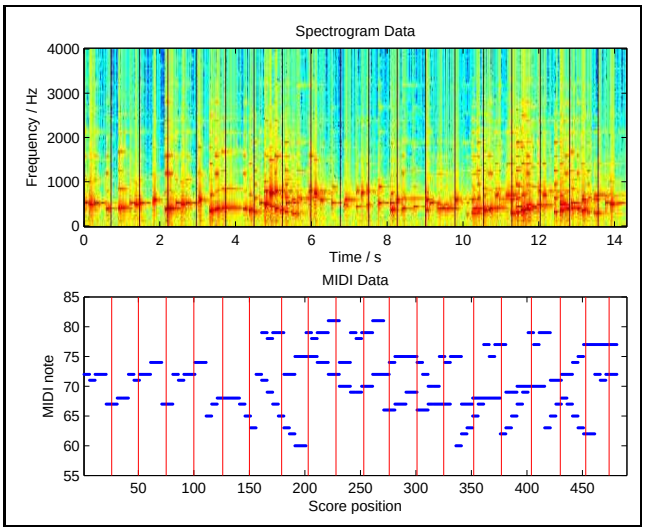


**Figure 6**. Viterbi-path score matching using the spectrogram model, $a = 10000$. 25 ms frames. Evenly spaced vertical bars in the spectrogram correspond to the score positions marked on the MIDI data. The variation in the spacing of the score positions illustrates the changing tempo through the extract.

downsampled to 8 kHz, from the RWC Musical Instrument Sound Database [7, 6]. The data in Figures 2 and 4 were obtained from these samples. We demonstrate offline score matching on the extract in Figure 5. The mp3 audio was downsampled to 8 kHz and divided into non-overlapping frames of 20ms length. Note that although in the score extract there are only two parts playing at a given time, when we sample the MIDI up to four simultaneous notes may be playing due neighboring notes overlapping in time. This is typical in *legato* piano playing and thus considering this overlap results in a robust score alignment. A quantization model describing the rendering of MIDI from score would need to take this effect into account. The score pointer transition probabilities $\pi(t_k)$ are chosen as $\{0.1, 0.3, 0.5\}$ for the tempo values { 'fast', 'medium', 'slow' } respectively.

Figures 6 and 7 are snapshots of the score alignment system, showing the position of the score pointer in time with respect to the observation data for the spectrogram and damped sinusoid models respectively. Animations of these figures are available on our website [2] , with the audio

[2] `http://www-sigproc.eng.cam.ac.uk/~php23/publications/ISMIR/`

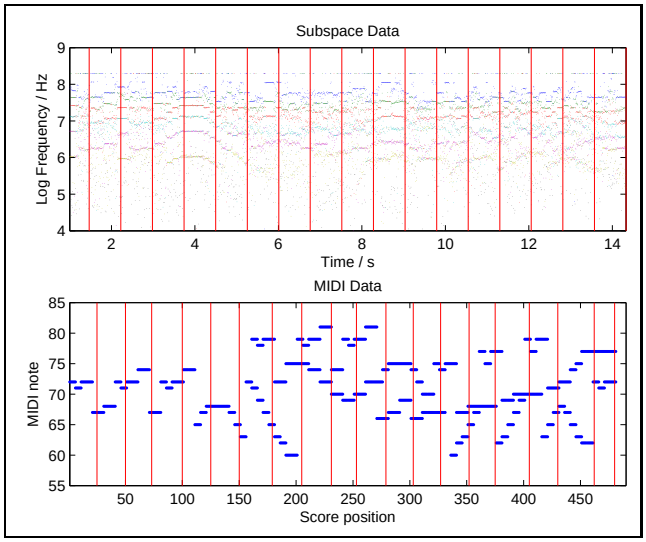


**Figure 7**. Viterbi-path score matching using the damped sinusoid model, $M = 7$. 7.5 ms frames. Only the frequency data from the subspace detector is shown.

signal playing simultaneously, from which it is clear that score pointer is correctly aligned with the audio signal. The damped sinusoid model gives better time-accuracy than the spectrogram model, although this comes at a significant computational overhead. This is because the subspace method gives high-resolution frequency estimates, while the spectrogram method returns frequency estimates in discrete bins, the resolution of which worsens with shorter frames.

## 6 CONCLUSIONS AND FUTURE WORK

In this paper we have considered a musical performance as the evolution of a score pointer over time. We have defined a simple dynamical model that governs the probability of the pointer transitioning from one position in the score to another. We have introduced two models of frequency-domain representations of musical audio given the set of notes present at the current score position. The first method is a generative model of spectrogram values, which is computationally efficient and simple to train on note samples, but has the usual time-frequency resolution limitation associated with the spectrogram. The second method models the output of a subspace detector, which returns the frequencies and amplitudes of a chosen number damped sinusoids in a frame. This method has better frequency resolution over shorter frame lengths, but is computationally more intensive. We have demonstrated these models in a score-alignment application, with promising results even for simple models linking frames together so that exact inference is possible.

Based on our results we suggest two possible applications of interest to the music information retrieval community. The computation requirements of the generative spectrogram model are sufficiently low to expect that a real-time score-following system would be feasible with

this model. The damped sinusoid model is capable of matching a score to audio with high time precision. With the training data and models used in this paper, we were able to detect note onsets to a resolution of 7.5 ms. This method could potentially be used to automatically annotate databases of audio where the score is known, with a high level of accuracy. Such databases are invaluable to researchers working on audio onset detection and music transcription wishing to evaluate the performance of their methods against ground truth.

We are also currently investigating other interesting applications that can be formulated in this framework such as audio synchronization, score-guided source separation or transcription.

## 7 REFERENCES


[1] R. Badeau, R. Boyer, and B. David. EDS parametric modeling and tracking of audio signals. In *Proceedings of the 5th International Conference on Digital Audio Effects*, Hamburg, Germany, September 2002.

[2] A. T. Cemgil, H. J. Kappen, and D. Barber. Generative model based polyphonic music transcription. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, October 2003.

[3] A. T. Cemgil, H. J. Kappen, and D. Barber. A generative model for music transcription. *IEEE Transactions on Audio, Speech and Language Processing*, 14(2):679–694, March 2006.

[4] S. Dixon. Live tracking of musical performances using on–line time warping. In *Proceedings of the 8th International Conference on Digital Audio Effects*, 2005.

[5] A. Doucet, J. F. G. de Freitas, and N. J. Gordon, editors. *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, 2000.

[6] M. Goto. Development of the RWC Music Database. In *Proceedings of the 18th International Congress on Acoustics*, volume 1, pages 553–556, April 2004. Invited Paper.

[7] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC Music Database: Music Genre Database and Musical Instrument Sound Database. In *Proceedings of the 4th ISMIR*, pages 229–230, October 2003.

[8] L. Grubb. *A Probabilistic Method for Tracking a Vocalist*. PhD thesis, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA, 1998.

[9] H. Heijink, P. Desain, H. Honing, and L. Windsor. Make me a match: An evaluation of different approaches to score-performance matching. *Computer Music Journal*, 24(1):43–56, 2000.

[10] T. Hoshishiba, S. Horiguchi, and I. Fujinaga. Study of expression and individuality in music performance using normative data derived from MIDI recordings of piano music. In *4th International Conference on Music Perception and Cognition*, pages 465–470, McGill University, Faculty of Music, Montreal, 1996.

[11] N. Hu, R. B. Dannenberg, and G. Tzanetakis. Polyphonic audio matching and alignment for music retrieval. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pages 185–188, New York, USA, 2003.

[12] M. I. Jordan, Z. Ghahramani, T. Jaakkola, and L. K. Saul. An introduction to variational methods for graphical models. *Machine Learning*, 37(2):183–233, 1999.

[13] K. Kashino and S.J. Godsill. Bayesian estimation of simultaneous musical notes based on frequency domain modelling. In *Proceedings of the International Conference on Acoustics, Speech and Signal Processing*, volume 4, pages 305–308, May 2004.

[14] E. W. Large. Dynamic programming for the analysis of serial behaviors. *Behavior Research Methods, Instruments, and Computers*, 25(2):238–241, 1993.

[15] M Muller, H Mattes, and F Kurth. An efficient multiscale approach to audio synchronization. In *Proceedings of the 6th International Conference on Music Information Retrieval*, Victoria, Canada, 2006.

[16] P. H. Peeling, C. Li, and S. J. Godsill. Poisson point process modeling for polyphonic music transcription. *Journal of the Acoustical Society of America Express Letters*, pages EL168–EL175, April 2007.

[17] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. In *Proceedings of the IEEE*, volume 22, pages 257–286, February 1989.

[18] C. Raphael. A hybrid graphical model for aligning polyphonic audio with musical scores. In *Proceedings of the 5th International Conference on Music Information Retrieval*, Barcelona, Spain, 2004.

[19] F. Soulez, X. Rodet, and D. Schwarz. Improving polyphonic and poly–instrumental music to score alignment. In *Proceedings of the 4th International Conference on Music Information Retrieval*, Baltimore, MD, 2003.

[20] N. Whiteley, A. T. Cemgil, and S. J. Godsill. Bayesian modelling of temporal structure in musical audio. In *Proceedings of the 7th International Conference on Music Information Retrieval*, Victoria, Canada, 2006.