BAYESIAN ESTIMATION OF SIMULTANEOUS MUSICAL NOTES BASED ON FREQUENCY DOMAIN MODELLING

*Kunio Kashino** *and* *Simon J. Godsill***

* NTT Communication Science Laboratories
3–1 Morinosato-Wakamiya, Atsugi,
243–0198, Japan.
kunio@eye.brl.ntt.co.jp

** University of Cambridge
Trumpington Street, Cambridge,
CB2 1PZ, U.K.
sjg@eng.cam.ac.uk

ABSTRACT

This paper proposes a Bayesian method for polyphonic music description. The method first divides an input audio signal into a series of sections called snapshots, and then estimates parameters such as fundamental frequencies and amplitudes of the notes contained in each snapshot. The parameter estimation process is based on a frequency domain modelling and Gibbs sampling. Experimental results obtained from audio signals of test note patterns are encouraging; the accuracy is better than 80 % for the estimation of fundamental frequencies in terms of semitones and instrument names when the number of simultaneous notes is two.

1. INTRODUCTION

This paper proposes a Bayesian method for polyphonic music description, where description means estimating parameters, such as fundamental frequencies, amplitudes, and instrument names, for the musical notes in an input sound signal. It is assumed that the input sound can be a monaural signal containing multiple simultaneous musical notes. Applications of the description we have in mind includes a "queries by polyphonic music" task in music information retrieval [1, 2].

Recently, many researchers have addressed music information retrieval based on audio signals. Their methods can be classified into two groups according to their objectives. One aims at retrieval of segments that are almost the same as the query by a signal level comparison. This approach has recently been referred to as audio fingerprinting. In this case, audio signal features, such as the spectrum feature, are used for the search [3]. The other group aims at retrieving segments "similar" to the query in some way, such as a segment having the same melody or a rearrangement of the music. It is difficult to accomplish this task solely by signal similarity, because re-takes, tempo variations, rearrangements, and many other factors alter the audio signals of music pieces considerably even if they are the same music in a title. Therefore, a certain description extracted from an audio signal is needed.

A typical method in the latter group is "query by humming" based on melody similarity. For example, Ghias *et al.* proposed monophonic melody matching using a melodic contour that represents the melody as strings of relative pitch ('U', 'D', and 'S') and showed its effectiveness [4]. So far, many related works have also been based on monophonic melody matching. However, this approach is not directly applicable to the task of polyphonic music retrieval by polyphonic music queries.

On the other hand, a considerable number of works targeting automatic music transcription, music scene analysis, or music scene description have been reported [5, 6, 7]. However, polyphonic music transcription is still a challenging task. Specifically, recognising multiple notes existing at the same time in a monaural signal is a difficult problem.

Here, we address estimation of multiple notes from the viewpoint of Bayesian modelling. This is because the Bayesian approach allows us to address the problem in terms of a unified probabilistic framework. In the music domain, only a few Bayesian approaches can be found in the literature [8, 9]. Of particular relevance to this current work is the one by Godsill and Davy, where a time-domain Bayesian harmonic model for musical pitch estimation was proposed [9]. In this paper, notes are modelled in the frequency domain to reduce computational cost, and to provide a simple framework to incorporate music-level statistical information, such as note or chord transition probabilities, into the processing.

Section 2 first describes a method to divide an input audio signal into processing windows, and then presents the parameter estimation method performed on each processing window. Section 3 shows experimental results. Section 4 gives conclusions.

2. METHOD

2.1. Processing window (snapshot) extraction

This section overviews the idea of SNAPs (Simultaneous-Note-set Alteration Points) and the "*snapshot*", which were proposed by Nagano, Kashino, and Murase [10].

A SNAP is a point in time when at least one note begins in a music audio signal. That is, a SNAP is the time when

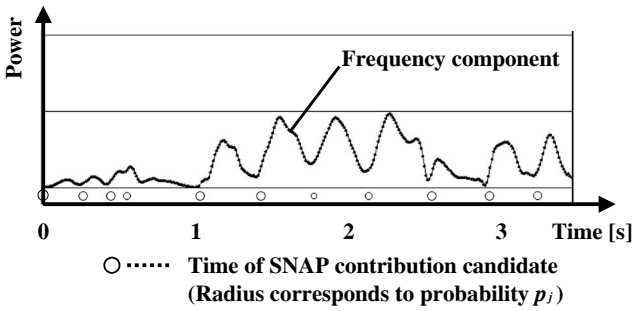

Figure 1: A frequency component and SNAP-CCs

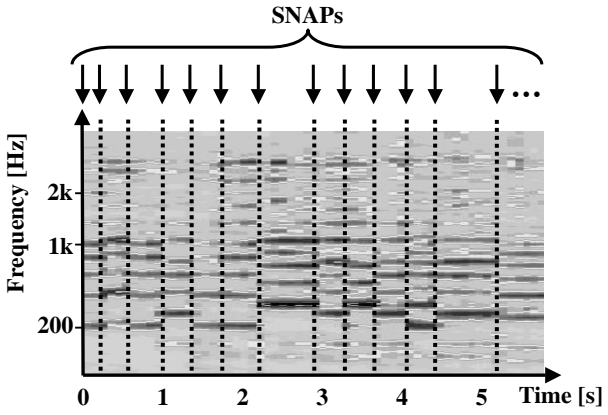

Figure 2: A music spectrogram and SNAPs

a set of sounding notes changes according to onset of a new note. We refer to the section between the adjacent SNAPs as a snapshot. The snapshot serves as a processing window for the succeeding parameter estimation process. Note that SNAP is similar to the idea of beat, but they are different in that generally there can be no (or multiple successive) onsets in one beat section. We introduce SNAP because our objective is to obtain a processing window for the estimation process rather than tracking musical beats.

SNAPs are extracted in the following process.

(1) Perform frequency analysis on an input signal to obtain a spectrogram. In our experimentation, a bandpass filter bank equally-spaced on a log frequency axis is employed for this purpose.

(2) Extract frequency components. A frequency component is a series of spectral local peaks on the spectrogram. The extraction is done by connecting spectral local peaks along the time axis.

(3) Extract SNAP contribution candidates (SNAP-CCs). A SNAP-CC is a point on a frequency component where the power forms a local minimum [6]. In addition, the starting point of a frequency component is always a SNAP-CC. Fig. 1 shows a freqency component and the times of SNAP-CCs.

(4) Calculate SNAP probability P_i for all SNAP-CCs using the following equation:

$$P_i = 1 - \prod_{j \in \Gamma_i} \left(1 - p_j \exp\left(-\frac{(t_i - t_j)^2}{\Delta T_i^2}\right)\right), \qquad (1)$$

where p_j is the probability that point j is the onset of a note (p_j can be estimated depending on the power variations of the frequency component), t_i is a time of i, and Γ_i is a set of SNAP-CCs in the temporal vicinity of i (*e.g.* within a 5 second time difference). ΔT_i is 1/8 of the fundamental beat interval. Here, the fundamental beat interval is the most frequent SNAP-CC interval extracted using a histogram of the SNAP-CC intervals.

(5) Calculate the total adjacent SNAP probability for each SNAP-CC by calculating the sum of SNAP probabilities of SNAP-CCs in the vicinity (*e.g.* within a 0.5-second time difference) of the SNAP-CC.

(6) Estimate the SNAP candidates, which are points on the time axis, by thresholding with respect to the maximum SNAP probability supporting each of the total adjacent SNAP probability.

(7) Eliminate spurious SNAPs from the SNAP candidates. Spurious SNAPs arise due to subtle onset asynchrony between simultaneous multiple notes or multiple frequency components. Elimination is done by thresholding with respect to intervals between SNAP candidates. That is, the following steps are repeated until all the SNAP candidates are processed: first we choose the SNAP candidate with the highest total adjacent SNAP probability, and then eliminate all the SNAP candidates existing in the temporal vicinity (*e.g.* 1/8 of the most frequent SNAP candidate intervals).

An example of SNAPs and the corresponding spectrogram extracted from a polyphonic music excerpt are shown in Fig. 2.

2.2. Mixing parameter estimation

The next step involves finding fundamental frequencies and amplitudes of the notes contained in each snapshot. For such purposes, various techniques, such as comb filtering, harmonic clustering [6], and MAP estimation [11] have been employed in the literature. Our approach here is based on a probabilistic framework.

Firstly, we model the observed spectrum containing multiple simultaneous notes in a snapshot as a linear combination of note spectra $x_n(\omega)$:

$$y(\omega) = \sum_{n=1}^{N} i_n a_n x_n(\omega) + v(\omega), \qquad (2)$$

where ω is frequency, $y(\omega)$ the observed sound spectrum, a_n the amplitude for the n-th template, i_n the binary (0/1) "indicator" term for the n-th template, and $v(\omega)$ is an additive noise term. In this paper, we refer to $i_n, a_n, x_n(\omega)$ as mixing parameters. We assume that x_n are stored in

advance and N is the number of the stored notes. We also assume $y(\omega)$ and $x_n(\omega)$ are real power spectra rather than complex ones.

Then, the problem we address here is to estimate mixing parameters, given the observation of spectrum $y(\omega)$ and the prior distributions of the parameters.

The posterior distribution for the mixing parameters may be written as

$$P(\boldsymbol{i}, \boldsymbol{a}, \boldsymbol{x}(\omega)|y(\omega)) \propto P(y(\omega)|\boldsymbol{i}, \boldsymbol{a}, \boldsymbol{x}(\omega))P(\boldsymbol{i})P(\boldsymbol{a})P(\boldsymbol{x}(\omega)), \quad (3)$$

where $\boldsymbol{i}$, $\boldsymbol{a}$, and $\boldsymbol{x}$ stand for the n dimensional vectors i_n, a_n, and x_n, respectively. Hereafter, we will omit ω for simplicity of notations where explicit expression is unnecessary.

Priors can be given considering the sequence of snapshots. Here, we consider a basic case where each snapshot is treated independently of each other. Then, for priors, we assume:

$$P(y|\boldsymbol{i}, \boldsymbol{a}, \boldsymbol{x}) = \mathrm{N}(\sum_{n=1}^{N} i_n a_n x_n, \sigma_v^2) \quad (4)$$

$$p(\boldsymbol{i}) = \text{independent Bernoulli priors}, \quad (5)$$

$$p(\boldsymbol{a}) = \mathrm{N}(\boldsymbol{\mu}_a, \boldsymbol{\Sigma}_a), \quad (6)$$

$$p(\boldsymbol{x}) = \mathrm{N}(\boldsymbol{\mu}_x, \boldsymbol{\Sigma}_x), \quad (7)$$

$$p(v) = \mathrm{N}(\mu_v, \sigma_v^2), \quad (8)$$

where $\mathrm{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes Gaussian distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$.

Then, Eq. (3) leads to:

$$p(i_k|i_{-k}, \boldsymbol{a}, \boldsymbol{x}, y) \propto \frac{b}{b+c}, \quad (9)$$

$$p(\boldsymbol{a}|\boldsymbol{i}, \boldsymbol{x}, y) \propto \mathrm{N}(\boldsymbol{\Phi}_a^{-1}\boldsymbol{\Theta}_a,\ \boldsymbol{\Theta}_a^{-1}), \quad (10)$$

$$p(\boldsymbol{x}|\boldsymbol{a}, \boldsymbol{i}, y) \propto \mathrm{N}(\boldsymbol{\Phi}_x^{-1}\boldsymbol{\Theta}_x,\ \boldsymbol{\Theta}_x^{-1}), \quad (11)$$

where i_{-k} denotes $\boldsymbol{i}$ except for i_k, and

$$b = P(\boldsymbol{a}, \boldsymbol{x}, y|i_{-k}, i_k = 1)P(i_{-k}, i_k = 1), \quad (12)$$

$$c = P(\boldsymbol{a}, \boldsymbol{x}, y|i_{-k}, i_k = 0)P(i_{-k}, i_k = 0), \quad (13)$$

$$\boldsymbol{\Phi}_a = \boldsymbol{\Sigma}_a^{-1} + \frac{\sum_\omega \boldsymbol{x}(\omega)\boldsymbol{x}^T(\omega)}{\sigma_v^2}, \quad (14)$$

$$\boldsymbol{\Theta}_a = \boldsymbol{\Sigma}_a^{-1}\boldsymbol{\mu}_a + \frac{\sum_\omega y(\omega)}{\sigma_v^2}\boldsymbol{x}(\omega), \quad (15)$$

$$\boldsymbol{\Phi}_x = \boldsymbol{\Sigma}_x^{-1} + \frac{\boldsymbol{a}\boldsymbol{a}^T}{\sigma_v^2}, \quad (16)$$

$$\boldsymbol{\Theta}_x = \boldsymbol{\Sigma}_x^{-1}\boldsymbol{\mu}_x + \frac{y}{\sigma_v^2}\boldsymbol{a}. \quad (17)$$

Eqs. (9)–(11) enable us to estimate expectations for $\boldsymbol{i}$, $\boldsymbol{a}$, $\boldsymbol{x}$ by Gibbs sampling. That is, those parameter values are obtained by drawing samples one by one from the distributions in the right-hand side of Eqs. (9)– (11) and then calculating the sample means [12].

The overall estimation procedure is as follows:

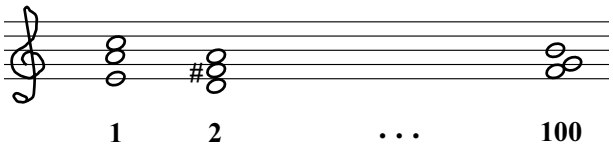

Figure 3: Test pattern example

1. In each snapshot, observe $y(\omega)$.
2. Set initial values of parameters $\boldsymbol{i}$, $\boldsymbol{a}$, and $\boldsymbol{x}(\omega)$.
3. Run the Gibbs sampling process described above.
4. Upon convergence, output the estimated parameter values and return to 1.

3. EXPERIMENTS

The objective of the experiments was to evaluate the basic characteristics of the proposed estimation method. Thus, we used audio test signals composed for this purpose. The test signal was a monaural multiple-simultaneous-notes pattern, as shown in Fig. 3. To create the pattern, we first recorded single notes of natural musical instruments (flute and violin) performed by professional players at a recording studio and stored the waveforms on a computer (16 bit, 48 kHz). We then randomly mixed the stored waveforms on a computer, selecting a designated number of notes. This means that each snapshot was independent of each other. The number of simultaneous notes was either two or three, while the number was not given to the parameter estimation process. The pitch range was between C_3 (523 Hz) and C_4 (1026 Hz).

The system was given $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$, which represent the statistical information of the templates. These parameters were obtained from another set of recordings of the musical instrument sounds (flute, violin, and piano). For each instrument, we recorded two samples for each of three different expressions (forte, normal, piano) for each semitone over the predetermined pitch ranges. We then calculated $\boldsymbol{\mu}_x$ and $\boldsymbol{\Sigma}_x$ for each instrument and semitone. That is, each of the templates $(x_1, x_2, \cdots, x_N)$ corresponds to a specific semitone of a specific instrument in this experimentation. Throughout the experiments, frequency analysis was done using 48 band-pass filters equally spaced on a log-frequency axis between 300 Hz and 4800 Hz.

The results are shown in Table 1. Case A is that when only the flute and violin templates between C_3 and C_4 were used ($N = 26$). Case B is that when the flute, violin and piano templates between C_3 and C_4 were used ($N = 39$). The recognition rate R was defined as

$$R = \max_\theta \left(\frac{p+r}{2}\right) \quad (18)$$

where θ is a threshold value for the obtained expectation values of $\boldsymbol{a}$. The precision rate p and the recall rate r were

Table 1: Random pattern results (R values)

Input	Case A	Case B
fl + vl Simul. 2 notes	90 %	80 %
fl + vl Simul. 3 notes	73 %	67 %

Case A: template={fl, vl}, $N = 26$
Case B: template={fl, vl, pf}, $N = 39$
fl: flute, vl: violin, pf: piano

defined as

$$p = \frac{(\#\text{correctly recognised notes})}{(\#\text{output notes in total})}, \tag{19}$$

$$r = \frac{(\#\text{correctly recognised notes})}{(\#\text{notes which should be recognised})}, \tag{20}$$

where "correct" means that both the fundamental frequency in terms of semitones and the instrument name are correct. The results listed in Table 1 show that the proposed estimation method works reasonably well. Especially, encouraging is that the accuracy was better than 80 % when the number of simultaneous notes was two, considering that the number of simultaneous notes was not given to the system in advance.

4. CONCLUSIONS

This paper has proposed a Bayesian method for mixing parameter estimation for musical sounds. The method firstly decomposes an input audio signal into sections called snapshots based on the detected onset candidates of notes, and then estimates parameters, such as fundamental frequencies and amplitudes of the notes contained in each snapshot. Unlike existing methods, our parameter estimation method is based on the Bayesian framework in the frequency domain, and thus we expect that musical knowledge (statistical information such as tendencies in note or chord transitions) can be straightforwardly incorporated as priors. The experiments described in this paper have focused on the validity of the parameter estimation mechanism, and future work will include evaluation using ordinary music performances. We also plan to apply the method to music information retrieval tasks.

5. ACKNOWLEDGMENTS

The authors wish to thank Drs. Noboru Sugamura, Shigeru Katagiri, Shoji Makino, and Hidehisa Nagano for their help and encouragement.

6. REFERENCES

[1] Nagano H., Kashino K., and Murase H.: "Fast Music Retrieval Using Polyphonic Binary Feature Vectors", *Proc. of ICME*, vol.1, pp.101–104 (2002).

[2] Pickens J., Bello J., Monti G., Crawford T., Dovey M., Sandler M., and Byrd D.: "Polyphonic Score Retrieval Using Polyphonic Audio Queries: A Harmonic Modeling Approach", *Proc. of ISMIR*, pp.140–149 (2002).

[3] Kashino K., Kurozumi T., and Murase H.: "A Quick Search Method for Audio and Video Signals Based on Histogram Pruning", *IEEE Trans. Multimedia*, vol.5, no.3, pp.348-357 (2003).

[4] Ghias A., Logan J., Chamberlin D., and Smith B. C.: "Query By Humming: Musical Information Retrieval in an Audio Database", *Proc. of ACM Multimedia*, pp.231–236 (1995).

[5] Sterian A. and Wakefield G. H.: "Music Transcription Systems: From Sound to Symbol", *Proc. of AAAI-2000 Workshop on AI and Music* (2000).

[6] Kashino K., Nakadai K., Kinoshita T., and Tanaka H.: "Organization of Hierarchical Perceptual Sounds: Music Scene Analysis with Autonomous Processing Modules and a Quantitative Information Integration Mechanism", *Proc. of IJCAI*, pp.158–164 (1995).

[7] Goto M.: "A Real-time Music Scene Description System: Detecting Melody and Bass Lines in Audio Signals", *Working Notes of the IJCAI-99 Workshop on Computational Auditory Scene Analysis*, pp.31–40 (1999).

[8] Kashino, K., Nakadai, K., Kinoshita, T., and Tanaka H.: "Application of Bayesian Probability Network to Music Scene Analysis", In *"Computational Auditory Scene Analysis"*, Lawrence Erlbaum Associates, pp.115–137 (1998).

[9] Godsill S. J. and Davy M.: "Bayesian Harmonic Models for Musical Pitch Estimation and Analysis", *Proc. of ICASSP* (2002).

[10] Nagano H., Kashino K., and Murase H.: "Similar Music Retrieval Using Polyphonic Binary Feature Vectors and Its Acceleration", *Trans. of IEICE*, D-II, vol.J86-D-II, no.11 (2003), *in Japanese*.

[11] Goto M.: "A Predominant-F0 Estimation Method for CD Recordings: MAP Estimation using EM Algorithm for Adaptive Tone Models", *Proc. of ICASSP*, pp.3365–3368 (2001).

[12] Gilks W. R., Richardson S., and Spiegelhalter D. J. (eds.): "Markov Chain Monte Carlo in Practice", *Chapman & Hall* (1996).