# SCALE- AND RHYTHM-AWARE MUSICAL NOTE ESTIMATION FOR VOCAL F0 TRAJECTORIES BASED ON A SEMI-TATUM-SYNCHRONOUS HIERARCHICAL HIDDEN SEMI-MARKOV MODEL

**First Author**
Affiliation1
author1@ismir.edu

**Second Author**
**Retain these fake authors in submission to preserve the formatting**

**Third Author**
Affiliation3
author3@ismir.edu

## ABSTRACT

This paper presents a statistical method that estimates a musically-natural sequence of musical notes from a vocal F0 trajectory. Since the onset times and F0s of sung notes are considerably deviated from the tatums and pitches indicated in a musical score, a score model is crucial for improving time-frequency quantization of F0s. We thus propose a hierarchical hidden semi-Markov model (HSMM) that combines a score model representing the rhythms and pitches of musical notes under musical scales with an F0 model representing the time-frequency deviations of F0s from the score. In the score model, musical scales are generated stochastically and note pitches are then generated according to the scales. Additionally, note onsets following a Markov process defined on the tatum grid are generated. In the F0 model, onset temporal deviations, smooth note-to-note F0 transitions, and F0 fluctuations are generated stochastically and added to the score. Given an F0 trajectory, our method estimates the most likely sequence of musical notes while giving more importance on the score model than the F0 model. Experimental results showed that the proposed method outperformed an HMM-based method having no models of scales and rhythms.

## 1. INTRODUCTION

Singing voice analysis is important for music information retrieval because a singing voice usually forms a large part of the melody line of popular music, and provides much information about music. Singing voice analysis techniques such as vocal F0 estimation [1,3,7,9,14] and singing voice separation [8, 12] have actively been studied and applied to singer identification [10, 22], karaoke generation [19], query-by-humming [8], and active music listening [6]. To leverage musical information conveyed by singing voices, it is necessary to convert a vocal F0 trajectory to a musical score containing only discrete symbols.

In this study, we tackle musical note estimation for a singing voice. This problem is challenging because a vocal F0 trajectory has considerable deviations from the corresponding musical score. The pitches and onset times of musical notes in a musical score can take only discrete values, whereas a vocal F0 trajectory is a continuous signal that can dynamically and smoothly vary over time. For example, the vocal F0 trajectory is modulated by vibrato and changes smoothly from one musical note to another by a portamento. These time-frequency deviations cause a standard transcription system to yield a sequence of musical notes that are inconsistent with music theory.

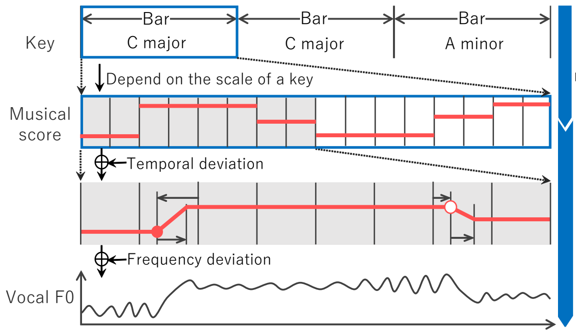



**Figure 1**: The generative process of a vocal F0 trajectory based on a hierarchical hidden semi-Markov model that combines a score model and an F0 model.

To solve this problem, we propose a statistical method of musical note estimation based on a hierarchical hidden semi-Markov model (HHSMM) that combines a score model for the process generating a sequence of musical notes and an F0 model for the process generating an F0 trajectory from the note sequence (Fig. 1). A key advantage of the proposed method is that a sequence of musical notes that sound more natural in terms of music theory can be obtained by using musical scales and rhythms as constraints on quantization of vocal F0 trajectories. In the score model, the semitone-level pitch of each note is determined depending on a scale that the note belongs to and the pitch of a previous note. On the other hand, the onset position of a note depends on that of a previous note so that the sequence of notes has appropriate rhythms. Then, the time-frequency fluctuations are added to a step-function-like F0 trajectory exactly corresponding to a musical score. Given a vocal F0 trajectory and tatum information, the scales, musical notes, and deviations, which are latent variables of the proposed model, are jointly estimated by using a Markov chain Monte Carlo algorithm.

## 2. RELATED WORK

In this section, we introduce related work on the analysis of singing voices.

### 2.1 Vocal F0 Estimation for Music Audio Signals

Estimation of vocal F0 trajectories for music audio signals has actively been studied [1, 3, 7, 9, 14], and the outputs of these methods can be used as inputs of our method. One of the most basic method is subharmonic summation (SHS) [7] that calculates the sum of the harmonic components of each candidate F0. Ikemiya *et al.* [9] improved F0 estimation based on SHS and singing voice separation based on robust principle component analysis (RPCA) [8] by using the mutual dependency of those two tasks. Salmon *et al.* [21] estimated contours of the melody F0 candidates by calculating a salience function and then recursively removed contours which do not form a melody line by using the characteristics of each contour. Durrieu *et al.* [3] extracted a main melody by representing accompaniments with a model inspired by non-negative matrix factorization (NMF) and leading voices with a source-filter model. Mauch *et al.* [14] modified the YIN [1] in a probabilistic way so that the modified system could determine multiple candidate fundamental frequencies and then choose one at each frame by using an HMM.

### 2.2 Musical Note Estimation for Singing Voices

Estimation of musical notes from sung melody have been a hot research topic [6, 11, 13, 15, 17, 18, 20, 23]. A naive method is to take the majority of vocal F0s in each interval of a regular grid [6]. Paiva *et al.* [17] proposed a step-by-step method with five stages: multipitch detection, multipitch trajectory construction, segmentation of multipitch trajectory, elimination of irrelevant notes, and extraction of notes that form a main melody. Raphael [18] proposed an HMM-based method that estimates pitches, rhythms, and tempos when the number of notes is given. The rhythm and onset deviation models used in [18] are similar to those used in our method. Laaksonen *et al.* [11] divided audio data into segments corresponding to keys and notes by focusing on the boundaries of chords given as input, and independently estimated the notes based on a score function. Ryynänen *et al.* [20] proposed a method based on a hierarchical HMM in order to capture the different kinds of vocal fluctuations (e.g., vibrato and portamento) within one note. In this model, the transition between pitches is represented in the upper-level HMM and the transition between the vocal fluctuations is represented in the lower-level HMM. Molina *et al.* [15] focused on the hysteresis characteristics of vocal F0s. Nishikimi *et al.* [16] proposed a method based on an HHM that represents the generative process of a vocal F0 trajectory considering the time and frequency deviations. Yang *et al.* [23] proposed a method based on a hierarchical HMM that represents the generative process of the $f_0$-$\Delta f_0$ plane. Mauch *et al.* [13] developed a software tool called Tony for extracting pitches. In this tool, a vocal F0 trajectory is estimated by PYIN [14], and musical notes are estimated by a modified version of Ryynänen's method [20].

## 3. PROPOSED METHOD

This section explains the proposed method of estimating a sequence of musical notes from a vocal F0 trajectory. The method is based on an HHSMM (Fig. 1) that stochastically generates the F0 trajectory with time-frequency deviations from a sequence of musical notes depending on musical scales. The upper part of the proposed model is an HMM that stochastically generates a sequence of musical notes according to the scales of keys that are assigned to bars. The lower part is an HSMM that represents the musical notes and temporal deviations as latent variables and the frequency deviations as F0 emission probabilities.

### 3.1 Problem Specification

The problem we tackle is defined as follows:

---

**Input**: A vocal F0 trajectory $\boldsymbol{X}=\{x_t\}_{t=1}^{T}$ and 16th-note-level tatums $\boldsymbol{Y}=\{(u_n, v_n)\}_{n=0}^{N}$,
**Output**: A sequence of notes $\boldsymbol{Z}=\{z_j=(p_j, l_j)\}_{j=0}^{J}$,

---

where $T$ is the number of frames in a vocal F0 trajectory, $x_t$ is a log frequency at time $t$, and $N$ is the number of 16th-note-level tatums. $u_n \in \{1, \ldots, T+1\}$ is the time of tatum $n$ and the beginning and end of music are represented as $u_0 = 1$ and $u_N = T+1$, respectively. $v_n \in \{0, \ldots, 15\}$ is the relative position of tatum $n$ in a bar. $J$ is the number of musical notes estimated by proposed methods, and the $j$-th note $z_j$ is represented as a pair consisting of an pitch $p_j \in \{1, \ldots, K\}$ and a note length $l_j \in \{1, \ldots, L\}$ in the unit of tatums, where $K$ is the number of kinds of semitone-level pitches, and $p_j$ indicates any one in $\{\mu_1, \ldots, \mu_K\}$, which is a set of log frequencies corresponding to semitone-level pitches. For convenience we introduce the initial note $z_0$ that does not appear in the actual score.

### 3.2 Probabilistic Modeling of Musical Scores

This section describes the score model representing rhythms and pitches of musical notes under musical scales.

#### 3.2.1 Modeling Key Transitions

Keys are represented as $\boldsymbol{S} = \{s_m\}_{m=0}^{M}$, where $M$ denotes the number of bars in the musical piece and $s_m$ denotes the key at the $m$-th bar. For convenience, we introduce the initial bar $s_0$ to which the initial note $z_0$ belongs. Instead of fixing one key for the whole piece, the key is allowed to change at bar lines. Each key $s_m$ takes one of the 24 values of $\{\text{C}, \text{C\#}, \cdots, \text{B}\} \times \{\text{major}, \text{minor}\}$. The latent variables $\boldsymbol{S}$ are described by a Markov chain as

$$p(s_0|\boldsymbol{\pi}) = \pi_{s_0}, \tag{1}$$

$$p(s_m|s_{m-1}, \boldsymbol{\xi}_{s_{m-1}}) = \xi_{s_{m-1}s_m}, \tag{2}$$

where $\boldsymbol{\pi}\in\mathbb{R}_{\geq 0}^{24}$ is a set of initial probabilities and $\boldsymbol{\xi}_s\in\mathbb{R}_{\geq 0}^{24}$ is a set of transition probabilities.

#### 3.2.2 Modeling Pitch Transitions

The sequence of pitches $\boldsymbol{P}$ is generated by a Markov chain depending on keys $\boldsymbol{S}$ as follows (Fig. 2):

$$p(p_0|s_0, \boldsymbol{\phi}_{s_0}) = \phi_{s_0 p_0}, \tag{3}$$

$$p(p_j|p_{j-1}, s_m, \boldsymbol{\psi}_{s_m p_{j-1}}) = \psi_{s_m p_{j-1} p_j}, \tag{4}$$

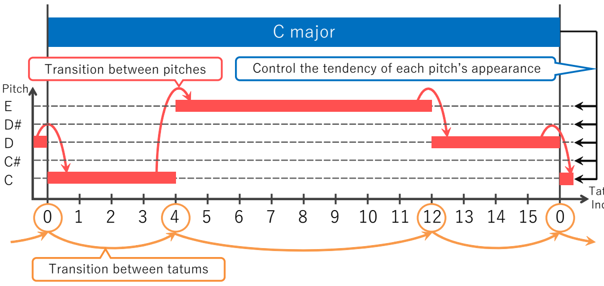



**Figure 2**: Overview of the score model.

where $\boldsymbol{\phi}_s \in \mathbb{R}^K_{\geq 0}$ is a set of initial probabilities, $\boldsymbol{\psi}_{sp} \in \mathbb{R}^K_{\geq 0}$ is a set of transition probabilities, and $m$ is the index of a bar to which the note $z_j$ belongs. Moreover, $\phi_{s_0 p_0}$ and $\psi_{s_m p_{j-1} p_j}$ are defined as

$$\phi_{s_0 p_0} = \frac{\hat{\phi}_{\hat{s}_0 \deg(p_0; s_0)}}{\sum_{p=1}^K \hat{\phi}_{\hat{s}_0 \deg(p; s_0)}}, \tag{5}$$

$$\psi_{s_m p_{j-1} p_j} = \frac{\hat{\psi}_{\hat{s}_m \deg(p_{j-1}; s_m) \deg(p_j; s_m)}}{\sum_{p=1}^K \hat{\psi}_{\hat{s}_m \deg(p_{j-1}; s_m) \deg(p; s_m)}}, \tag{6}$$

where $\hat{s} \in \{\text{major}, \text{minor}\}$ is the mode of key $s$ and $\deg(p;s) \in \{0, \ldots, 11\}$ is the degree of pitch $p$ in key $s$ (defined as the relative pitch class of $p$ from the tonic of key $s$). $\hat{\phi}_*$ and $\hat{\psi}_*$ are the initial and transition probabilities of pitch classes, given the scales.

#### *3.2.3 Modeling Onset Transitions*

Considering the transition between onset positions of adjacent notes, the model makes $\boldsymbol{Z}$ have the plausible rhythm. Let $r_{j-1} \in \{v_n\}_{n=1}^N$ be the onset position of the $j$-th note $z_j$. The transition probability is given by

$$p(r_j | r_{j-1}, \boldsymbol{\zeta}_{r_{j-1}}) = \zeta_{r_{j-1} r_j}, \tag{7}$$

where the distance between $r_{j-1}$ and $r_j$ indicates the note value $l_j$ of note $z_j$. We assume that $r_0 = v_0$ and $r_J = v_N$.

### 3.3 Probabilistic Modeling of F0 Trajectories

The section describes the F0 model that represents the time-frequency deviations of F0s from the musical score.

#### *3.3.1 Modeling Temporal Deviations*

We assume that vocal F0 trajectories include the following two types of temporal deviations (Fig. 3a):

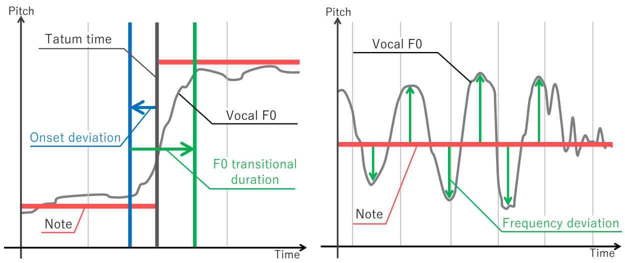
**Onset deviation**: the gap between the vocal onset time and the note onset time.

**F0 transitional duration**: the time it takes for singing voices to finish transitioning from one pitch to the next.

The onset deviations $\boldsymbol{G} = \{g_j\}_{j=0}^J$ accompanying with $\boldsymbol{Z}$ are represented as discrete latent variables. Each $g_j$ can take an integer value between $-G$ and $G$. As with the onset position model, $g_{j-1}$ denotes the onset deviation of note $z_j$. We assume that each $g_j$ is independently generated by

$$p(g_j | \boldsymbol{\rho}) = \rho_{g_j}, \tag{8}$$

where $\boldsymbol{\rho} \in \mathbb{R}^{2G+1}_{\geq 0}$ is a set of onset deviation probabilities. We assume that there are no deviations for the onset of the first note and the offset of the last note, *i.e.*, $g_0 = g_J = 0$.




(a) Temporal deviations (b) Frequency deviations

**Figure 3**: Deviations in a vocal F0 trajectory.

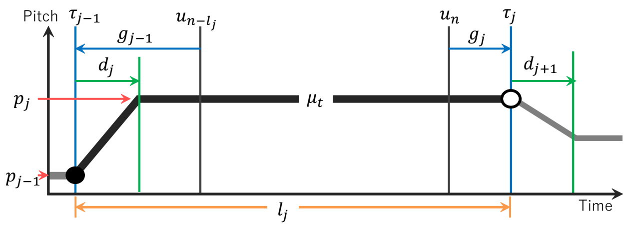



**Figure 4**: The black bold line represents a sequence of the location parameters of the Cauchy distributions.

The F0 transitional durations $\boldsymbol{D} = \{d_j\}_{j=1}^J$ accompanying with $\boldsymbol{Z}$ are also represented as discrete latent variables. Each $d_j$ can take a value from 1 to $D$. The continuous transition of vocal F0s between notes $z_{j-1}$ and $z_j$ is represented by a slanted line spanning $d_j$ frames. We assume that each $d_j$ is independently generated as follows:

$$p(d_j | \boldsymbol{\eta}) = \eta_{d_j}, \tag{9}$$

where $\boldsymbol{\eta} \in \mathbb{R}^D_{\geq 0}$ is a set of duration probabilities.

#### *3.3.2 Modeling Frequency Deviations*

The vocal F0 trajectory $\boldsymbol{X}$ is generated by imparting probabilistic frequency deviations to the sequence of notes to which temporal deviations have already been imparted (Fig. 3b). Assuming that $x_t$ is independently generated at each frame, the emission probability of the $j$-th note $z_j$ is given by

$$\begin{aligned} & p(x_{\tau_{j-1}:\tau_j - 1} | p_{j-1}, p_j, l_j, g_{j-1}, g_j, d_j, \hat{\mu}_t, \lambda) \\ & = \prod_{t=\tau_{j-1}}^{\tau_j - 1} \{\delta_{x_t, \text{voiced}} \text{Cauchy}(x_t | \hat{\mu}_t, \lambda) + \delta_{x_t, \text{unvoiced}}\} \\ & = e_{p_{j-1} p_j l_j g_{j-1} g_j d_j}, \end{aligned} \tag{10}$$

where $X_{\tau':\tau-1}$ indicates $x_{\tau'}, \ldots, x_{\tau-1}$, $\lambda$ is a scale parameter, and $\hat{\mu}_t$ (Fig. 4) is a location parameter given by

$$\hat{\mu}_t = \begin{cases} \frac{\mu_{p_j} - \mu_{p_{j-1}}}{d_j}(t - \tau_{j-1}) + \mu_{p_{j-1}} & (\tau_{j-1} \leq t < \tau_j + d_j) \\ \mu_{k_j} & (\tau_{j-1} + d_j \leq t < \tau_j) \end{cases}. \tag{11}$$

When the onset of note $z_{j+1}$ is located at the $n$-th tatum, $\tau_j = u_n + g_j$ and $\tau_{j-1} = u_{n-l_j} + g_{j-1}$.

### 3.4 Prior Distributions

We put conjugate Dirichlet priors on categorical model parameters $\boldsymbol{\pi}$, $\boldsymbol{\xi}$, $\hat{\boldsymbol{\phi}}$, $\hat{\boldsymbol{\psi}}$, $\boldsymbol{\zeta}$, $\boldsymbol{\rho}$, and $\boldsymbol{\eta}$ as follows:

$$\boldsymbol{\pi}_s \sim \text{Dirichlet}(\boldsymbol{a}^\pi_s), \quad \boldsymbol{\xi}_{sk} \sim \text{Dirichlet}(\boldsymbol{a}^\xi_{sp}),$$

$$\hat{\boldsymbol{\phi}}_{\hat{s}} \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\hat{\phi}}_{\hat{s}}\right),\ \ \hat{\boldsymbol{\psi}}_{\hat{s}\deg(p;s)} \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\hat{\psi}}_{\hat{s}\deg(p;s)}\right),$$
$$\boldsymbol{\zeta}_r \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\zeta}_r\right),$$
$$\boldsymbol{\rho} \sim \mathrm{Dirichlet}(\boldsymbol{a}^{\rho}),\ \ \boldsymbol{\eta} \sim \mathrm{Dirichlet}(\boldsymbol{a}^{\eta}), \quad (12)$$

where $\boldsymbol{a}^{\pi}\in\mathbb{R}^{26}_{+}$, $\boldsymbol{a}^{\xi}_{s}\in\mathbb{R}^{26}_{+}$, $\boldsymbol{a}^{\hat{\phi}}_{\hat{s}}\in\mathbb{R}^{12}_{+}$, $\boldsymbol{a}^{\hat{\psi}}_{\hat{s}\deg(p;s)}\in\mathbb{R}^{12}_{+}$, $\boldsymbol{a}^{\zeta}_{r}\in\mathbb{R}^{16}_{+}$, $\boldsymbol{a}^{\rho}\in\mathbb{R}^{2G+1}_{+}$, and $\boldsymbol{a}^{\eta}\in\mathbb{R}^{D}_{+}$ are hyperparameters.

Since the Cauchy distribution does not have a conjugate prior, we put a Gamma prior on $\lambda$ as

$$\lambda \sim \mathrm{Gamma}\left(a^{\lambda}_0, a^{\lambda}_1\right), \quad (13)$$

where $a^{\lambda}_0$ and $a^{\lambda}_1$ are shape and rate hyperparameters.

## 3.5 Bayesian Inference

Given an F0 trajectory $\boldsymbol{X}$, we aim to calculate the posterior distribution $p(\boldsymbol{Q}, \boldsymbol{S}, \boldsymbol{\Theta}|\boldsymbol{X})$, where $\boldsymbol{Q} = \{\boldsymbol{P}, \boldsymbol{L}, \boldsymbol{G}, \boldsymbol{D}\}$ (latent variables) and $\boldsymbol{\Theta} = \{\boldsymbol{\pi}, \boldsymbol{\xi}, \hat{\boldsymbol{\phi}}, \hat{\boldsymbol{\psi}}, \boldsymbol{\zeta}, \boldsymbol{\rho}, \boldsymbol{\eta}\}$ (model parameters). Since this calculation is analytically intractable, we use Markov chain Monte Carlo (MCMC) methods. To get samples of the latent variables $\boldsymbol{S}$ and $\boldsymbol{Q}$, forward filtering-backward sampling algorithms are used. To get samples of $\boldsymbol{\Theta}$ except for $\lambda$, a set of parameters with conjugate priors, a Gibbs sampling algorithm is used. Since there is no conjugate prior for the parameter $\lambda$, we use the Metropolis-Hastings (MH) algorithm. Since $\boldsymbol{S}$ and $\boldsymbol{Q}$ share the sequence of notes $\boldsymbol{Z}$ and are mutually dependent, each variable is updated as follows:

1. Initialize notes $\boldsymbol{Z}$ with a majority-vote method.
2. Update the sequence of keys $\boldsymbol{S}$ based on given $\boldsymbol{Z}$.
3. Update $\boldsymbol{Q}$ based on given $\boldsymbol{S}$.
4. Update the model parameters $\boldsymbol{\Theta}$.
5. Return to 2.

### *3.5.1 Inferring Latent Variables S*

Given the sequence of notes $\boldsymbol{Z}$, each $s_m$ is sampled in accordance with the probability given by

$$\beta^{S}_{s_m} = p(s_m|s_{m+1:M}, \boldsymbol{Z}), \quad (14)$$

where $s_{m+1:M}$ represents $s_{m+1}, \ldots, s_M$. The calculation of Eq. (14) and sampling of keys $\boldsymbol{S}$ are performed by the forward filtering-backward sampling method.

In forward filtering, we recursively calculate the probability $\alpha^{S}_{s_m}$ as follows:

$$\alpha^{S}_{s_0} = p(p_0, s_0) = p(p_0|s_0)p(s_0) = \phi_{s_0 p_0}\pi_{s_0}, \quad (15)$$
$$\alpha^{S}_{s_m} = p(p_{0:j_{m+1}-1}, s_m)$$
$$= \prod_{j=j_m}^{j_{m+1}-1} \psi_{s_m p_{j-1} p_j} \sum_{s_{m-1}} \xi_{s_{m-1} s_m} \alpha^{S}_{s_{m-1}}, \quad (16)$$

where $j_m$ is the index of the first note whose onset belongs to the $m$-th bar. $j_m$ can be calculated from given note values $\boldsymbol{L}$.

In backward sampling, Eq. (14) is calculated by using the values calculated in forward filtering, and keys are sampled recursively as follows:

$$\beta^{S}_{s_M} = p(s_M|\boldsymbol{Z}) \propto \alpha^{S}_{s_M}, \quad (17)$$
$$\beta^{S}_{s_m} = p(s_m|s_{m+1:M}, \boldsymbol{Z}) \propto \alpha^{S}_{s_m}\xi_{s_m s_{m+1}}. \quad (18)$$

### *3.5.2 Inferring Latent Variables Q*

The latent variables $\boldsymbol{Q}$ can be estimated in a way similar to that in which the latent variables $\boldsymbol{S}$ are inferred. In forward filtering, we recursively calculate the probability $\alpha^{Q}_{p_n l_n, g_n d_n}$ as follows:

$$\alpha^{Q}_{p_0 l_0 g_0 d_0} = p(p_0|\boldsymbol{S}) = \phi_{y_0 p_0}, \quad (19)$$
$$\alpha^{Q}_{p_n l_n g_n d_n} = p(x_{1:\tau_n - 1}, p_n, l_n, g_n, d_n|\boldsymbol{S})$$
$$= \begin{cases} 0 & (l_n>n) \\ \rho_{g_n}\eta_{d_n}\zeta_{r_n r_0} \\ \quad \cdot \sum_{p_0} \psi_{s_1 p_0 p_n} e_{p_0 p_n l_n 0 g_n d_n} \alpha^{Q}_{p_0 l_0 g_0 d_0} & (l_n=n) \\ \sum\limits_{p_{n'}, g_{n'}} \sum\limits_{l_{n'}}^{\min(n',L)} \sum\limits_{d_{n'}} \rho_{g_n}\eta_{d_n}\zeta_{r_n r_{n'}}\psi_{s_{m(n')} p_{n'} p_n} \\ \quad \cdot e_{p_{n'} p_n l_n g_{n'} g_n d_n} \alpha^{Q}_{p_{n'} l_{n'} g_{n'} d_{n'}} & (l_n<n) \end{cases}, \quad (20)$$

where $\tau_n = u_n + g_n$, $n' = n - l_n$, and $m(n')$ is the index of the bar that the $n'$-th tatum belongs to. $p_n$, $l_n$, $g_n$, and $d_n$ are the variables of forward messages that correspond to the note whose offset position is at the $n$-th tatum $u_n$. Note that these variables are different from $j$-indexed variables $p_j$, $l_j$, $g_j$, and $d_j$. Since the onset and offset times of the note $z_n = (p_n, l_n)$ are respectively the $(n-l_n)$-th tatum and the $n$-th tatum, the probability $p(l_n)$ which appears in the recursive calculation of Eq. (20) is replaced by $p(r_n|r_{n-l_n})$.

In backward sampling, the posterior distribution of the latent variables is calculated by using the values calculated in forward filtering, and notes and temporal deviations are sampled recursively as follows:

$$\beta_{p_N l_N g_N d_N} = p(p_N, l_N, g_N, d_N|\boldsymbol{X}, \boldsymbol{S}) \propto \alpha^{Q}_{p_N l_N g_N d_N},$$
$$\beta_{p_{n'} l_{n'} g_{n'} d_{n'}}$$
$$= p(p_{n'}, l_{n'}, g_{n'}, d_{n'}|p_{n:N}, l_{n:N}, g_{n:N}, d_{n:N}, \boldsymbol{X})$$
$$\propto \begin{cases} 0 & (l_n>n) \\ e_{p_{n'} p_n l_n g_{n'} g_n d_n}\psi_{s_{m(n')} p_{n'} p_n} \\ \quad \cdot \zeta_{r_{n'} r_n}\rho_{g_n}\eta_{d_n}\alpha^{Q}_{p_{n'} l_{n'} g_{n'} d_{n'}} & (l_n \le n) \end{cases}. \quad (21)$$

### *3.5.3 Learning Model Parameters Θ*

The posterior distributions of the model parameters with the prior distributions are calculated using $\boldsymbol{S}$ and $\boldsymbol{Q}$ obtained in the backward sampling steps, and these parameters are sampled according to the posterior distributions as follows:

$$\boldsymbol{\pi} \sim \mathrm{Dirichlet}(\boldsymbol{a}^{\pi} + \boldsymbol{b}^{\pi}), \quad (22)$$
$$\boldsymbol{\xi}_s \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\xi}_{y} + \boldsymbol{b}^{\xi}_{s}\right), \quad (23)$$
$$\hat{\boldsymbol{\phi}}_{\hat{s}} \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\hat{\phi}}_{\hat{s}} + \boldsymbol{b}^{\hat{\phi}}_{\hat{s}}\right), \quad (24)$$
$$\hat{\boldsymbol{\psi}}_{\hat{s}\deg(p;s)} \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\hat{\psi}}_{\hat{s}\deg(p;s)} + \boldsymbol{b}^{\hat{\psi}}_{\hat{s}\deg(p;s)}\right), \quad (25)$$
$$\boldsymbol{\zeta}_r \sim \mathrm{Dirichlet}\left(\boldsymbol{a}^{\zeta}_{r} + \boldsymbol{b}^{\zeta}_{r}\right), \quad (26)$$
$$\boldsymbol{\rho} \sim \mathrm{Dirichlet}(\boldsymbol{a}^{\rho} + \boldsymbol{b}^{\rho}), \quad (27)$$
$$\boldsymbol{\eta} \sim \mathrm{Dirichlet}(\boldsymbol{a}^{\eta} + \boldsymbol{b}^{\eta}), \quad (28)$$

where $\boldsymbol{b}^{\pi}\in\mathbb{R}_{\geq 0}^{26}$ is a unit vector whose $s_0$-th element is 1. $\boldsymbol{b}_s^{\xi}\in\mathbb{R}_{\geq 0}^{26}$ is a vector whose $s'$-th element indicates the number of transitions between adjacent keys $y$ and $y'$ in the sequence of latent variables $\boldsymbol{Y}$. $\boldsymbol{b}^{\rho}\in\mathbb{R}_{\geq 0}^{2G+1}$ is a vector whose $g$-th element indicates the number of vocal onset deviations of $g$ in sampled $\boldsymbol{Q}$, and $\boldsymbol{b}^{\eta}\in\mathbb{R}_{\geq 0}^{D}$ is a vector whose $d$-th element represents the number of F0 transitional durations of $d$ in sampled $\boldsymbol{Q}$. $\boldsymbol{b}_r^{\zeta}\in\mathbb{R}_{\geq 0}^{16}$ is a vector whose $r'$-th element represents the number of transitions between adjacent note onset positions $r$ and $r'$ in $\boldsymbol{R}=\{r_j\}_{j=0}^{J}$ that can be calculated from the note values $\boldsymbol{L}$ sampled in backward sampling. Regarding the vector $\boldsymbol{b}_{\hat{s}}^{\hat{\phi}}\in\mathbb{R}_{>0}^{12}$, when the key of the initial bar and the pitch of the initial note are $s_0 = s$ and $p_0 = p$, the value of the element $b_{\hat{s}\deg(p;s)}^{\hat{\phi}}$ is 1, and the other elements are 0. Regarding the vector $\boldsymbol{b}_{\hat{s}\deg(p;s)}^{\hat{\psi}}\in\mathbb{R}_{\geq 0}^{12}$, the value of $b_{\hat{s}\deg(p;s)\deg(p';s)}^{\hat{\psi}}$ is increased by one when there is a transition from a pitch $p$ to a pitch $p'$ under a key $s$ in the sampled latent variables.

To apply the MH sampling to the parameter $\lambda$, we define a random-walk proposal distribution as follows:

$$p(\lambda^*|\lambda) = \mathrm{Gamma}(\gamma\lambda, \gamma), \tag{29}$$

where $\lambda^*$ is a proposal, $\lambda$ is the current sample, and $\gamma$ is a hyperparameter. The proposal $\lambda^*$ is accepted as the next sample according to the probability given by

$$A(\lambda^*, \lambda) = \min\left\{\frac{\mathcal{L}(\lambda^*)p(\lambda|\lambda^*)}{\mathcal{L}(\lambda)p(\lambda^*|\lambda)}\right\}, \tag{30}$$

where

$$\mathcal{L}(\lambda) = \mathrm{Gamma}\big(\lambda|\phi_0^{\lambda}, \phi_1^{\lambda}\big)\prod_{j=0}^{J} e_{p_{j-1}p_j l_j g_{j-1} g_j d_j}, \tag{31}$$

and $\{p_j, l_j, g_j, d_j\}_{j=0}^{J}$ are the values sampled in the backward sampling. The value of $\lambda$ is updated by $\lambda^*$ only when the value of $A(\lambda^*, \lambda)$ is larger than a random number sampled from the uniform distribution $\mathcal{U}(0, 1)$.

### 3.6 Viterbi Decoding

The sequence of latent variables $\boldsymbol{S}$ and $\boldsymbol{Q}$ are estimated with the Viterbi algorithm with the model parameters that maximize the joint distribution $p(\boldsymbol{X}, \boldsymbol{Q}, \boldsymbol{S}, \boldsymbol{\Theta}|\boldsymbol{\Phi})$ in the learning process. As in the inference of latent variables, we initialize $\boldsymbol{Z}$ by the majority-vote method, $\boldsymbol{S}$ is estimated based on $\boldsymbol{Z}$, and then $\boldsymbol{Q}$ is estimated depending on the $\boldsymbol{S}$ estimated in the previous step.

In the Viterbi decoding on keys $\boldsymbol{S}$, the value $\omega_s^S$ is recursively calculated as follows:

$$\omega_{s_0}^S = \ln\phi_{s_0 k_0} + \ln\pi_{s_0}, \tag{32}$$

$$\omega_{s_m}^S = \sum_{j=j_m}^{j_{m+1}-1}\ln\psi_{s_m p_{j-1} p_j} + \max_{s_{m-1}}\left\{\ln\xi_{s_{m-1}s_m} + \omega_{s_{m-1}}^S\right\}. \tag{33}$$

In the recursive calculation of $\omega_s^S$, the previous state $s_{m-1}$ that maximizes the value of $\omega_{s_m}^S$ is memorized as $c_{s_m}^S$, and the keys $\boldsymbol{S}$ are recursively estimated as follows:

$$s_M = \arg\max_{s_M}\alpha_{s_M}^S, \tag{34}$$

$$s_{m-1} = c_{s_m}^S. \tag{35}$$

In the Viterbi decoding on variables $\boldsymbol{Q}$, the value $\omega_{plgd}^Q$ is recursively calculated as follows:

$$\omega_{p_0 l_0 g_0 d_0}^Q = \mathrm{w}^{\phi}\ln\phi_{s_0 p_0}, \tag{36}$$

$$\omega_{p_n l_n g_n d_n}^Q = \begin{cases} -\inf & (l_n > n) \\ \mathrm{w}^{\rho}\ln\rho_{g_n} + \mathrm{w}^{\eta}\ln\eta_{d_n} + \mathrm{w}^{\zeta}\ln\zeta_{r_n r_0} & \\ \quad + \max_{p_0}\Big\{\mathrm{w}^{\psi}\ln\psi_{s_1 p_0 p_n} & \\ \quad + \mathrm{w}^{e}\ln e_{p_0 p_n l_n 0 g_n d_n} + \omega_{p_0 l_0 g_0 d_0}^Q\Big\} & (l_n = n) \\ \mathrm{w}^{\rho}\ln\rho_{g_n} + \mathrm{w}^{\eta}\ln\eta_{d_n} + \mathrm{w}^{\zeta}\ln\zeta_{r_n r_{n'}} & \\ \quad + \max_{(p_{n'}, l_{n'}, g_{n'}, d_{n'})}\Big\{\mathrm{w}^{\psi}\ln\psi_{s_{m(n')} p_{n'} p_n} & \\ \quad + \mathrm{w}^{e}\ln e_{p_{n'} p_n l_n g_{n'} g_n d_n} + \omega_{p_{n'} l_{n'} g_{n'} d_{n'}}^Q\Big\} & (l_n < n) \end{cases}, \tag{37}$$

where $\mathrm{w}^{\phi}$, $\mathrm{w}^{\psi}$, $\mathrm{w}^{\rho}$, $\mathrm{w}^{\eta}$, $\mathrm{w}^{\zeta}$, and $\mathrm{w}^{e}$ are the weight parameters that control the balance between probabilities. In the recursive calculation of $\omega_{plgd}^Q$, the previous states $p_{n'}$, $l_{n'}$, $g_{n'}$, and $d_{n'}$ which maximize the value of $\omega_{p_n l_n g_n d_n}^Q$ are memorized as $c_{p_n l_n g_n d_n}^Q$, and the variables $\boldsymbol{Q}$ are recursively estimated as follows:

$$(p_N, l_N, g_N, d_N) = \arg\max_{p_N, l_N, g_N, d_N}\alpha_{p_N l_N g_N d_N}^Q, \tag{38}$$

$$(p_{n'}, l_{n'}, g_{n'}, d_{n'}) = c_{p_n l_n g_n d_n}^Q. \tag{39}$$

## 4. EVALUATION

We reports comparative experiments conducted to evaluate the performance of the proposed method in musical note estimation from vocal F0 trajectories.

### 4.1 Experimental Conditions

Among the 100 pieces of popular music in the RWC music database [5], we used 63 pieces s do not include 32nd notes, triplets, harmonizing parts, and overlaps of adjacent notes, which are not considered by the proposed method. The input F0 trajectories were obtained from the annotation data [4] or automatically estimated by using the state-of-the-art melody extraction method proposed in [9]. The annotation data contain unvoiced regions and the estimation data do not. The tatum times and onset positions were obtained from the annotation data.

The Bayesian inference and Viterbi decoding were independently conducted for each song. The onset transition probabilities were learned in advance from a corpus of rock music [2]. The hyperparameters were $\boldsymbol{a}^{\pi}=\mathbf{1}$, $\boldsymbol{a}_s^{\xi}=\mathbb{1}$, $\boldsymbol{a}_r^{\zeta}=\mathbf{1}$, $\boldsymbol{a}^{\rho} = \boldsymbol{a}^{\eta} = a_0^{\lambda} = a_1^{\lambda} = \gamma = 1$, where $\mathbb{1}$ and $\mathbf{1}$ respectively represent the matrix and vector whose elements are all ones. $\boldsymbol{a}_{\hat{s}}^{\hat{\phi}}$ and $\boldsymbol{a}_{\hat{s}\deg(p;s)}^{\hat{\psi}}$ are vectors in which the elements corresponding to musical notes on the scale of $\hat{s}$ are 1 and the others are 0.1. The weight parameters of the Viterbi algorithm were $\mathrm{w}^{\phi} = \mathrm{w}^{\psi} = 29.4$, $\mathrm{w}^{\rho} = 2.4$, $\mathrm{w}^{\eta} = 2.9$, $\mathrm{w}^{\zeta} = 48.5$, and $\mathrm{w}^{e} = 3.8$. To obtain musically-natural sequences of musical notes, we put more emphasis on the score model than the F0 model.

For comparison, we tested the majority-vote method as a baseline and the latest conventional method based on a

| Model | Input F0s | Tatum level | Note level |
|---|---|---|---|
| Proposed method | Ground-truth | $72.5 \pm 1.6$ | $28.3 \pm 2.1$ |
| | Estimated | $68.8 \pm 1.3$ | $30.9 \pm 1.7$ |
| With only rhythms | Ground-truth | $71.7 \pm 1.6$ | $26.9 \pm 2.0$ |
| | Estimated | $67.7 \pm 1.3$ | $29.1 \pm 1.8$ |
| With only scales | Ground-truth | $68.4 \pm 1.6$ | $11.5 \pm 1.3$ |
| | Estimated | $65.5 \pm 1.2$ | $13.7 \pm 1.1$ |
| Without scales & rhythms | Ground-truth | $67.6 \pm 1.5$ | $10.4 \pm 1.2$ |
| | Estimated | $64.6 \pm 1.2$ | $12.7 \pm 1.1$ |
| Majority vote | Ground-truth | $54.1 \pm 1.5$ | $20.1 \pm 1.4$ |
| | Estimated | $61.0 \pm 1.4$ | $22.0 \pm 1.5$ |
| HMM [16] | Estimated | $68.0 \pm 1.2$ | $14.8 \pm 1.3$ |

**Table 1**: Average matching rates [%] and their standard errors in tatum and note levels.

semi-beat-synchronous HMM [16]. To evaluate the effectiveness of the score model, we tested four versions of the proposed method; a method that does not consider scales (key transition probabilities) and rhythms (onset transition probabilities), a method considering only scales, a method considering only rhythms, the full method considering both scales and rhythms. To accelerate the inference, the search range of pitches was limited around the pitches estimated by the majority-vote method.

To evaluate the performance of each method, we calculated tatum-level and note-level matching rates by comparing the estimated sequences of musical notes with the ground-truth data. The tatum-level matching rate is the rate of the number of tatum units whose pitches were estimated correctly to the total number of tatum units whose pitches exist in the ground-truth scores. The note-level matching rate is the rate of the number of musical notes whose pitches, onsets, and offsets were estimated correctly to the total number of musical notes in the ground-truth scores. If adjacent notes in the ground-truth scores have the same pitch or are connected by a tie, those notes were regarded as a single note. Since the compared method [16] outputs a pitch in a 16th-note-wise manner, a sequence of the same pitches was regarded as a single note.

### 4.2 Experimental Results

The experimental results are shown in Table 1 [1] . The proposed method outperformed the majority-vote method and the conventional method in terms of both measures. Comparing the tatum-level matching rates obtained by the four versions of the proposed method, we confirmed that the score model significantly improved the performance of musical note estimation. The use of the onset transition probabilities (rhythm constraints) was found to be more effective than that of the key transition probabilities (scale constraints). Although the tatum-level matching rate obtained the proposed method (68.8%) was close to that obtained by the conventional method (68.0%), the note-level matching rate obtained the proposed method (30.9%) was significantly better than that obtained by the conventional method

[1] The results of music note estimation by the proposed method are available online: https://anonymous170428.github.io/

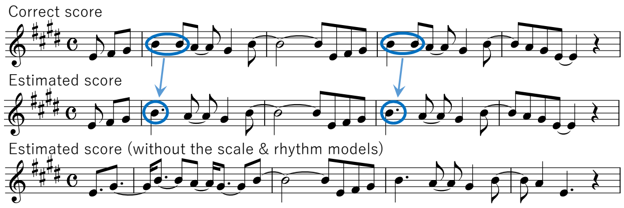



**Figure 5**: Musical scores estimated from a ground-truth F0 trajectory by the proposed method and its variant without scale and rhythm constraints.

(14.8%), This is a remarkable advantage of the proposed HHSMM that can directly represent both the pitches and durations (onsets and offsets) of musical notes on symbolic musical scores, not on continuous-time piano rolls.

Examples of estimated musical scores are illustrated in Fig. 5. The proposed method yielded the almost accurate musical score except that some notes were merged. To correctly recognize two adjacent notes with the same pitch, it is necessary to refer to original singing voices or music audio signals. The score estimated without considering the score model, on the other hand, included a lot of wrong notes that were inconsistent with music theory. This result also shows the effectiveness of using the score model as musical constraints on musical note estimation.

## 5. CONCLUSION

This paper presented a statistical method for musical note estimation from a vocal F0 trajectory. Our method is based on an HHSMM that combines a score model (HMM) representing the generative process of a musical score from musical scales with an F0 model (HSMM) representing the generative process of a vocal F0 trajectory with time-frequency deviation from the musical score. We confirmed that the proposed method can yield more musically-natural sequences of musical notes, which significantly improves the perceived quality of estimated results.

One of the most interesting directions of this research is to use the proposed model as a musically-meaningful prior distribution on a vocal F0 trajectory in vocal F0 estimation for music audio signals. We plan to integrate the proposed "language" model that generates an F0 trajectory from a musical score with an acoustic model that generates a spectrogram from the F0 trajectory in a hierarchical Bayesian manner. This enables us to jointly learn the vocal F0 trajectory and musical score from music audio signals. Joint estimation of beat times and F0s is worth investigating to overcome the problem of estimation-error accumulation in the cascaded estimation approach.

## 6. REFERENCES

[1] A. de Cheveigné and H. Kawahara. YIN, a fundamental frequency estimator for speech and music. *The Journal of the Acoustical Society of America*, 111(4):1917–1930, 2002.

[2] T. De Clercq and D. Temperley. A corpus analysis of rock harmony. *Popular Music*, 30(01):47–70, 2011.

[3] J.-L. Durrieu, G. Richard, B. David, and C. Févotte. Source/filter model for unsupervised main melody extraction from polyphonic audio signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 18(3):564–575, 2010.

[4] M. Goto. Aist annotation for the RWC music database. In *The 7th International Conference on Music Information Retrieval (ISMIR 2006)*, pages 359–360, 2006.

[5] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka. RWC music database: Popular, classical and jazz music databases. In *The 3rd International Conference on Music Information Retrieval (ISMIR 2002)*, pages 287–288, 2002.

[6] M. Goto, K. Yoshii, H. Fujihara, M. Mauch, and T Nakano. Songle: A web service for active music listening improved by user contributions. In *Proc. of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011)*, pages 311–316, 2011.

[7] Dik J. Hermes. Measurement of pitch by subharmonic summation. *The journal of the acoustical society of America*, 83(1):257–264, 1988.

[8] P.-S. Huang, S. D. Chen, P. Smaragdis, and M. Hasegawa-Johnson. Singing-voice separation from monaural recordings using robust principal component analysis. In *2012 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2012)*, pages 57–60. IEEE, 2012.

[9] Y. Ikemiya, K. Yoshii, and K. Itoyama. Singing voice analysis and editing based on mutually dependent F0 estimation and source separation. In *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2015)*, pages 574–578. IEEE, 2015.

[10] Y. E. Kim and B. Whitman. Singer identification in popular music recordings using voice coding features. In *3rd International Conference on Music Information Retrieval (ISMIR 2002)*, volume 13, page 17, 2002.

[11] A. Laaksonen. Automatic melody transcription based on chord transcription. In *Proc. of the 15th International Society for Music Information Retrieval (ISMIR 2014)*, pages 119–124, 2014.

[12] Y. Li and D. Wang. Separation of singing voice from music accompaniment for monaural recordings. *IEEE Transactions on Audio, Speech, and Language Processing*, 15(4):1475–1487, 2007.

[13] M. Mauch, C. Cannam, R. Bittner, G. Fazekas, J Salamon, J. Dai, J. Bello, and S Dixon. Computer-aided melody note transcription using the Tony software: Accuracy and efficiency. In *Proc. of the 1st International Conference on Technologies for Music Notation and Representation (TENOR 2015)*, pages 23–30, 2015.

[14] M. Mauch and S. Dixon. pYIN: A fundamental frequency estimator using probabilistic threshold distributions. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014)*, pages 659–663. IEEE, 2014.

[15] E. Molina, L. J. Tardón, A. M. Barbancho, and I. Barbancho. Sipth: Singing transcription based on hysteresis defined on the pitch-time curve. *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, 23(2):252–263, 2015.

[16] R. Nishikimi, E. Nakamura, K. Itoyama, and K Yoshii. Musical note estimation for f0 trajectories of singing voices based on a bayesian semi-beat-synchronous hmm. In *Proc. of the 17th International Society for Music Information Retrieval Conference (ISMIR 2026)*, pages 461–467, 2016.

[17] R. P. Paiva, T. Mendes, and A. Cardoso. On the detection of melody notes in polyphonic audio. In *6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 175–182, 2005.

[18] C. Raphael. A graphical model for recognizing sung melodies. In *6th International Conference on Music Information Retrieval (ISMIR 2005)*, pages 658–663, 2005.

[19] M. Ryynänen, T. Virtanen, J. Paulus, and A. Klapuri. Accompaniment separation and karaoke application based on automatic melody transcription. In *2008 IEEE International Conference on Multimedia and Expo*, pages 1417–1420. IEEE, 2008.

[20] M. P. Ryynänen and A. P. Klapuri. Automatic transcription of melody, bass line, and chords in polyphonic music. *Computer Music Journal*, 32(3):72–86, 2008.

[21] J. Salamon and E. Gómez. Melody extraction from polyphonic music signals using pitch contour characteristics. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(6):1759–1770, 2012.

[22] W.-H. Tsai and H.-M. Wang. Automatic singer recognition of popular music recordings via estimation and modeling of solo vocal signals. *IEEE Transactions on Audio, Speech, and Language Processing*, 14(1):330–341, 2006.

[23] L. Yang, A. Maezawa, J. B. L. Smith, and E. Chew. Probabilistic transcription of sung melody using a pitch dynamic model. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2017)*, pages 301–305. IEEE, 2017.