Metrical-accent aware vocal onset detection in polyphonic audio

April 20, 2017

Abstract

The goal of this study is the detection of onsets of singing voice in polyphonic audio recordings. Starting with a hypothesis that the knowledge of the current position in a metrical cycle can improve the accuracy of vocal note onset detection, we propose a novel probabilistic model to jointly track the beats and vocal note onsets. The proposed model extends a state of the art probabilistic model for beat and meter tracking, in which a-priori probability of a note at a specific position in the metrical cycle (i.e. metrical accent) interacts with the probability of observing a vocal note onset. We carried out evaluation on a varied collection of multi-instrument datasets from two music traditions (English popular music and Turkish makam) with different types of metrical cycles and singing styles. Results confirm that the proposed model reasonably improves vocal note onset detection accuracy compared to a baseline model that does not take metrical position into account.

1 Introduction

The automatic detection of vocal onsets is usually performed in the context of singing voice transcription.

In this work we propose a novel probabilistic model that tracks simultaneously accents in a metrical cycle and note onsets of singing voice. We assume that an onset feature is emitted that represents a metrical accent. Also simultaneously a vocal pitch value is emitted, from which vocal note onsets can be inferred. The vocal *Vs* non-vocal detection is done as a preprocessing step. The knowledge if a frame is vocal or not is integrated in the pitch observation model.

2 Related Work

2.1 Singing voice transcription

In the case of monophonic singing the more successful approaches are based on discontinuities in pitch contours [8]. The idea is usually referred to as 'island building' groups temporally candidate onsets and has been the basis for other works.

A probabilistic note HMM is presented in [10], where a note has 3 states: attack (onset), stable pitch state and silent state. The transition probabilities are learned from data. Later this approach has been extended by adding a separate HMM for regions of transition between the steady notes [5]. Recently Mauch et al. suggested to compact the musical knowledge into rules as a way to describe the observation and transition likelihoods, instead of learning them from data [7]. The authors cover a range with distinct pitch from lowest MIDI C2 up to B7. Each MIDI pitch is further divided into 3 sub-pitches, resulting in $n = 207$ notes with different pitch, each having the 3 note states A, S and N. Although being conceptually capable of tracking onsets in singing voice audio with accompaniments, these approaches were tested only on a cappella singing.

In the case when background instruments are present, essential first step is to extract the predominant pitch curve. In recent work a flamenco-specific approach has been suggested [6]. In flamenco some musical aspects evince significant deviations from the principles of Western music style. From this perspective the authors suggest a tradition-specific approach. As a primary step of the note transcription stage, notes are segmented by a set of flamenco-specific onset detection rules, based on pitch contour and volume characteristics. Varying size of the Gaussian filter, we showed in a previous study that using this method notes from Makam singing can be segmented with reasonable recall of 75 and 65 % in the monophonic and polyphonic case [Dzhambazov].

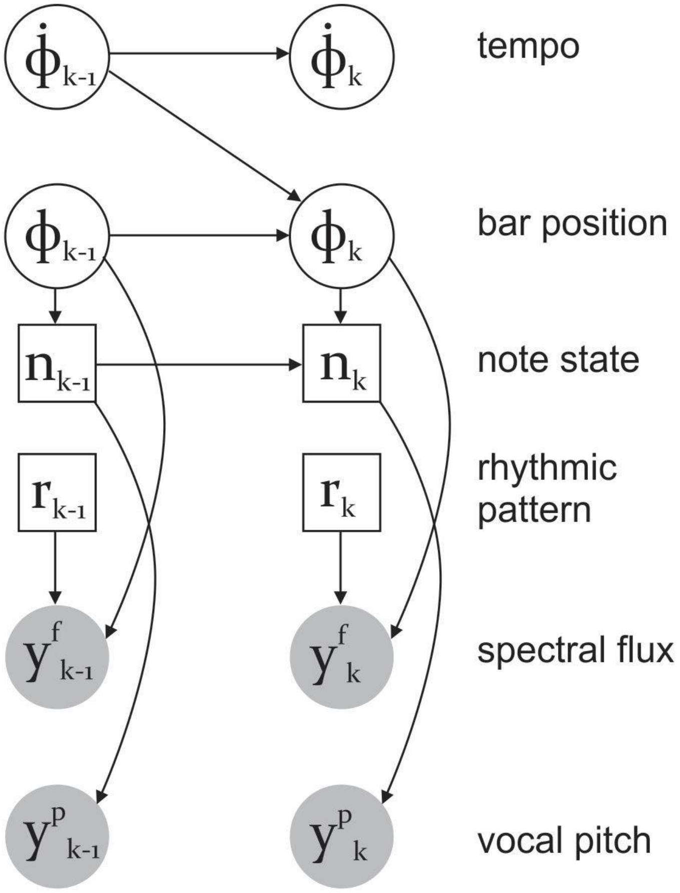

Figure 1: a dynamic Bayesian network for the simultaneous beat and note onset detection

2.2 Beat Detection

In [3] the authors present a probabilistic approach for meter inference from polyphonic music signals. It can track simultaneously tempo and the position in a metrical cycle. MORE

3 Approach

We base our model on the meter inference system presented in [3]. We modify it by adding a hidden state for vocal note, which depends on the current position in a metrical cycle. The model is represented as a DynamicBayesian Network (DBN) in Figure 1.

3.1 Hidden states

3.1.1 Tempo state $\dot{\phi}$ and bar position state ϕ

To assure feasible computational time we relied on the combined bar-tempo efficient state space, presented in [4]. The bar position ϕ represents.. The bar positions are discretized with different resolution at each different tempo. $\dot{\phi}$ encodes how much bar positions the pointer advances from the current to the next time instant. The total number of bar positions $|\phi|$ is derived by multiplying b by the number of beats in a metrical cycle. To keep the size of the bar-tempo state space small, we input the ground truth tempo for each recording, allowing $\dot{\phi}$ to deviate within ±10 bpm from it. Another motivation to limit the tempo in such a way is avoiding possible octave errors in the beat tracking, which would not be desirable for beat-aware note onset detection. This yields around 100-1000 states for the bar positions within a single beat and around 10 K in total.

3.1.2 Note state n_k

The note states are a modified version of these suggested in the note transcription model of [7]. We adopted the first two HMM note states: attack region (A), stable pitch region (S) and replaced the silent one with non-vocal state (N). Because full-fledged note transcription is outside the scope of this work, instead of 3 steps per semitone, we used for simplicity only a single one, which deteriorated just slightly the note onset detection accuracy. Also, to reflect the pitch range in the dataset, on which we evaluate, we set as minimal MIDI note E3 covering almost 3 octaves up to B5 (35 semitones). This totals to 105 note states.

3.1.3 Rhythmic pattern r_k

A rhythmic patterns describes the pattern of accents in a metric cycle. For simplicity we use only one rhythmic pattern for each metric cycle. Since the metrical type for each recording from the dataset is known a priori, a hidden state for the rhythm pattern is not modeled explicitly.

To be able to represent the DBN as a hidden Markov model, the bar-tempo efficient state space is combined with the note state space into a joint state space x. The joint state space is a cartesian product of the two state spaces, resulting in up to 10 K x 105 = 1 M states.

3.2 Transition model

The tempo transition probability $p(\dot{\phi}_k|\dot{\phi}_{k-1})$ and bar position probability $p(\phi_k|\phi_{k-1})$ are the same as in [4]. Transition from one tempo to another is allowed only at the change of beat. This is a reasonable assumption for local tempo deviations the analyzed datasets, which can be considered to occur relatively beat-wise.

3.2.1 Note transition probability $p(n_k|n_{k-1})$

The probability of advancing to a next note state is based on the transitions of the note-HMM, introduced in [7]. Let use briefly review it: Within a note, except for the self-transitions, the only possibility is to progress to next state (see Figure). To ensure continuity each of the self-transition probabilities is rather high: for example the one for non-vocal state P_{nn} is set to 0.9999, which is constant for all notes. Let $P_{n'a''}$ be the probability of transition from non-vocal state $N_{l'}$ after note l' to attack state $A_{l''}$ of its following note l''. The authors assume that small pitch changes are more likely than larger ones, approximately according to a normal distribution centered at pitch change of zero ([7], Figure 1.b). Now we can rewrite the note transition of [7] as:

$$p(n_k|n_{k-1}) = \begin{cases} P_{n'a''} & n_{k-1} = N_{l'}, \quad n_k = A_{l''} \\ P_{nn} & n_{k-1} = n_k = N \end{cases} \tag{1}$$

Note also that for transitions from non-vocal states it should hold that $\sum_{a''} p_{n'a''} = 1 - P_{nn}$.

In this study, we modify $1 - P_{nn}$ to vary in time depending on the current bar position ϕ_k. Let b_k be the beat, closest in time to a current bar position ϕ_k.

$$1 - P_{nn} = 0.0001 + \Theta(\phi_k)e(b_k) \tag{2}$$

where

$\Theta(\phi_k)$: function weighting the contribution a beat adjacent to current bar position ϕ_k

$e(b)$: probability of a note onset co-occurring with the b^{th} beat in the metrical cycle (b ∈ (0,|B|))

We propose two weighting functions for two scenarios: when beats are manually annotated and when they are automatically detected

Manually annotated beats Although many notes are aligned with a beat in the musical score, in performance singers often advance or delay slightly note onsets in relation to the beats. In addition, manual beat annotations are not perfect and often have a slight time bias. To reflect these facts, we introduce a normal distribution $N_{0,\sigma}$ to smooth the importance of beats over a time window around a beat:

$$\Theta(b_k) = [N_{0,\sigma}(d(\phi_k, b_k))]^w \tag{3}$$

where

w : sensitivity of note onset probability to beats

$d(\phi_k, b_k)$: the distance from current bar position ϕ_k to closest beat position b_k

Equation 3 means essentially, that $p_{n'a''}$ is kept, but scaled differently depending on how close in time to a beat it is. .

Automatically detected beats To keep the transition model light we decided to modify $1 - P_{nn}$ only at beat positions. Therefore the weighting function is set to 1 only at bar positions corresponding to beat positions, and zero elsewhere:

$$\Theta(\phi_k) = 1_{d(\phi_k, b_k)=0} \tag{4}$$

When beats are automatically detected simultaneously to onsets we expect that the system is more flexible and do not need a closest-beat weighting scheme.

3.3 Observation Models

The default hopsize of the predominant vocal detection $h_v = 0.0058$ seconds, whereas the one used for computing the spectral flux feature in [3] $h_s = 0.02$ seconds. Increasing h_v deteriorated the note onset accuracy reasonably. Therefore in this work we used hopsize of 0.0058 seconds.

3.3.1 Bar Observation model

In this paper for the bar position observation $P(y_k^f|x_k)$ we use the same observation model proposed in [3]. A Spectral Flux-like feature, y^f, which represents the strength of rhythmic onsets at different bar positions, is extracted from the audio signal.

3.3.2 Pitch Observation Model

We adopt the idea proposed in [7] that a vocal note state emits pitch y^p according to normal distribution, centered around the note pitch. The standard deviation for all stable states is set to ..., whereas the one of the attack states is ... The pitch contour is extracted with *PredominantMelodyMakam* - an algorithm for predominant pitch extraction, extended from [11] and tailored to Makam Music. As an output each frame k is assigned a pitch value and probability of being voiced v_k [1]. Simultaneously to pitch tracking *PredominantMelodyMakam* performs voicing detection. However, since correct vocal pitch is crucial for the sake of this study, we preferred to rely on manual vocal annotations and thus assigned $v_k = 0$ for all non-vocal frames.

For each state the observation probability $P(y_k^p|x_k)$ of vocal states is normalized to sum to v_k (unlike the original model which sums to a global constant v). This leaves the probability for each non-vocal state be $1 - v_k/n$.

3.4 Learning model parameters

3.4.1 Bar Observation model

To train the rhythmic patterns, per usul we set aside 6 recordings with beat annotations, which we did not use in evaluation. For each usul we trained one pattern by fitting a 2-GMM on the spectral-flux-like feature vector. Analogously to [3] we pooled the bar positions down to 16 patterns per beat. The feature vector was normalized to zero mean, unit variance and taking moving average. Normalization is done per song.

3.4.2 Probability of note onset

The probability of a vocal note onset co-occurring at a given bar position p_b is learned separately for each usul. We adopt the probabilities learned in [2].

3.5 Inference

The optimal hidden sequence of the joint state space $x_{1:K}$ can de estimated by multiplying the observation models:

$$P(X_{1:K}|Y^f_{1:K}, Y^p_{1:K}) = P(X_1)\Pi^K_{k=2}P(x_k|x_{k-1})P(y^f_k|x_k)P(y^p_k|x_k)$$

We obtain $x_{1:K}$ by decoding with the well-known Viterbi algorithm. A Beat is detected when the bar position variable hits one of B positions of strokes within the metrical cycle. A vocal note onset is detected when the state path enters an attack note state after being in non-vocal state.

Note that the size of the state space poses a memory requirement. A recording of 1 minute has around 10 K frames at $h_v = 0.0058$. To use Viterbi thus requires to store in memory pointers to up to 4 G states, which amounts to 40 G RAM (with uint32 data type).

4 Datasets

4.1 Turkish makam

The Turkish dataset is a subset of the dataset, presented in [3], including only recordings with singing voice present. It is divided into training and test parts. The test dataset comprises 5 1-minute excerpts from recordings with solo singing voice for each of three rhythm classes, referred to as usuls in OTMM: 9/8-usul Aksak, 10/8-usul Curcuna, and the 8/8-usul Düyek. All excerpts are manually annotated with beats, downbeats and vocal note onsets. The training set spans around 7 minutes of audio from each of the three usuls, annotated also manually with beats and downbeats. Due to the scarcity of material with solo singing voice, several excerpts with choir sections were included

https://docs.google.com/spreadsheets/d/1f9wyxB6emGHvVGUhIjNQwSxxhOJhPAdNzA2BieyKVXw/edit?usp=sharing

4.2 English pop songs

We selected 10 30-second clips of English pop songs, which have been aligned to their corresponding MIDIs in a recent study [9]. The audio recordings are 30-second thumbnails of the Million Song Dataset[1]. We generated the singing voice transcription from the vocal MIDI channel and the corresponding aligned timestamps for the audio. We added annotations of vocal segments.

5 Experiments

5.1 Evaluation metrics

Beat detection Since improvement of the beat detector is outside the scope of this study, we report accuracy of detected beats only in terms of their f-measure. This serves solely as reference to existing work[2]. The f-measure can take a maximum value of 1, while beats tapped on the off-beat relative to annotations will be assigned an f-measure of 0. We used the default tolerance window of 70 ms, also applied in [3].

Onset detection We measured note onset accuracy in terms of precision and recall. Unlike a cappella singing, the exact onset times of singing voice accompanied by instruments, might be much more ambiguous. To accommodate this fact, we adopted the tolerance of t=50 ms, used for vocal onsets in accompanied flamenco singing by [6]. Note transcription accuracy remains outside the scope of this study.

[1] http://colinraffel.com/projects/lmd/

[2] Note that f-measure is agnostic to the phase of the detected beats, which is clearly not optimal.

	metrical cycle	beat F-meas	P	R	F-meas
M	düyek	-	33.1	31.6	31.6
beat-aware		86.4	37.8	36.1	36.1
M	aksak	-	42.1	36.9	37.9
beat-aware		72.9	45.0	39.0	40.3
M	curcuna	-	30.2	26.2	26.8
beat-aware		39.6			
M	4/4	-	67.3	45.3	
beat-aware		73.3	67.1	47.2	
beat-aware	total (no curcuna)		40.9	37.3	37.9

Table 2: Average accuracy of detected vocal note onsets with tolerance window of 50 ms. Results are averaged per usul

5.2 Experiment 1: with manually annotated beats

As a precursor to evaluating the full-fledged model, we conducted an experiment with manually annotated beats. This is done to test the general feasibility of the proposed note transition probability 3.2.1, unbiased from errors in the beat detection. To this end, the detection of note onsets is carried out by a reduced model with a single hidden state: the note state. The observation model is reduced to the pitch observation model. To represent the dependence of the note transition probability on the closest beat, a variable-time HMM is implemented, similar to the one presented in [Dzhambazov, ISMIR2016].

We have tested different pairs of values for w and σ from Eq. 2. Onset detection accuracy peaked at w=1 and σ= 0.03 seconds. As a baseline we run the algorithm of [7] with the same number of note states, which is essentially the same note transition model with fixed $1 - P_{NN} = 0.0001$. Table 1 presents the absolute accuracy compared to the baseline. Inspection showed that the proposed model added some onsets around beats, which are missed by baseline. Note that in [7] the authors introduce a post-processing step, in which onsets of consecutive sung notes with same pitch are detected considering their intensity difference. We excluded this step in all system variant presented, because it could not be integrated in the proposed DBN model. This means that essentially in this work cases of consecutive same-pitch notes are missed, which decreases somewhat the recall compared to the original algorithm.

	P	R	F-meas
M	36.9	33.8	34.3
beat-aware	40.4	39.5	39.0

Table 1: Average accuracy of detected vocal note onsets with tolerance window of 50 ms with annotated beats. NO CURCUNA!. M stands for the baseline of Mauch et al.

5.3 Experiment 2: simultaneous beat and note detection

The average f-measure of detected beats per usul can be seen in Table 2[3]. Except for curcuna the beat tracking accuracy is comparable to the results reported in [3, Table 1.a-c, R=1] We also observe a reasonable improvement of not onset metrics in cases when the beats are detected rather reliably (for düyek and aksak).

Importantly, the note onset accuracy is almost the same as with manual beat annotations. In preliminary experiments we saw that using the transition with the lightweight weighting $\Theta(b_k)$ from Eq. 4 with manual beat annotations results in onset accuracy much worse than with the weighting presented in Eq. 3. This means that the simultaneous tracking of beats in a DBN contributes to a better-informed note onset detection compared to beats inout from a preprocessing step.

[3] per-recoding results can be found in sheet 2 of `https://docs.google.com/spreadsheets/d/1f9wyxB6emGHvVGUhIjNQwSxxhOJhPAdNzA2BieyKVXw/edit?usp=sharing`

6 Conclusions

TODO: Could be used for beat-infromed transcription of singing voice.

References

[1] Hasan Sercan Atlı, Burak Uyar, Sertan Şentürk, Barış Bozkurt, and Xavier Serra. Audio feature extraction for exploring Turkish makam music. In *3rd International Conference on Audio Technologies for Music and Media*, Ankara, Turkey, 2014. Bilkent University, Bilkent University.

[2] André Holzapfel. Relation between surface rhythm and rhythmic modes in turkish makam music. *Journal of New Music Research*, 44(1):25–38, 2015.

[3] Andre Holzapfel, Florian Krebs, and Ajay Srinivasamurthy. Tracking the" odd": Meter inference in a culturally diverse music corpus. In *ISMIR*, pages 425–430, 2014.

[4] Florian Krebs, Sebastian Böck, and Gerhard Widmer. An efficient state-space model for joint tempo and meter tracking. In *ISMIR*, pages 72–78, 2015.

[5] Willie Krige, Theo Herbst, and Thomas Niesler. Explicit transition modelling for automatic singing transcription. *Journal of New Music Research*, 37(4):311–324, 2008.

[6] Nadine Kroher and Emilia Gómez. Automatic transcription of flamenco singing from polyphonic music recordings. *IEEE/ACM Transactions on Audio, Speech and Language Processing*, 24(5):901–913, 2016.

[7] Matthias Mauch, Chris Cannam, Rachel Bittner, George Fazekas, Justin Salamon, Jiajie Dai, Juan Bello, and Simon Dixon. Computer-aided melody note transcription using the tony software: Accuracy and efficiency. In *Proceedings of the First International Conference on Technologies for Music Notation and Representation (TENOR 2015)*, pages 23–30, 2015.

[8] Rodger J McNab, Lloyd A Smith, and Ian H Witten. Signal processing for melody transcription. 1995.

[9] Colin Raffel. *Learning-Based Methods for Comparing Sequences, with Applications to Audio-to-MIDI Alignment and Matching*. PhD thesis, COLUMBIA UNIVERSITY, 2016.

[10] Matti Ryynänen. Probabilistic modelling of note events in the transcription of monophonic melodies. 2004.

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