MODEL FOR MEMORY-BASED MUSIC TRANSCRIPTION AND ITS VARIATIONAL BAYES SOLUTION

Štěpán Albrecht[1], *Václav Šmídl*[2]

[1]University of West Bohemia, Plzeň, Czech Republic, albrs@kiv.zcu.cz
[2] Institute of Information Theory and Automation, Prague, Czech Republic, smidl@utia.cas.cz

ABSTRACT

The problem of memory based music transcription is considered and a probabilistic model for polyphonic music is proposed. Parameters of the model correspond to labels of the pre-recorded sounds and their amplitudes. Since exact estimation of the parameters is computationally prohibitive, we develop an approximate estimation algorithm using the Variational Bayes approximation. Results of the proposed algorithm are compared to alternative algorithms on piano recordings

1. INTRODUCTION

Automatic music transcription (AMT) is a process of decomposing recorded music signal into a sequence of higher-level sound events. The entire AMT—i.e. resolving pitch, loudness, timing and instrument of all sound events in an input audio music signal [1]—is not theoretically possible in general [1], therefore practical AMT has to be restricted to a specific scenario. Commonly used scenarios are memory-based and data-based AMT. The former utilizes sound models corresponding to a certain musical instrument sound (allowing to identify the instruments), the latter utilizes only rules which hold in general. We are concerned with a special case of the memory-based AMT. Kashino's transcription system [2] is another example of memory-based AMT system in the sense of [1].

Intuitively, our formalization of the problem can be understood as an 'inverse music sequencer', Fig. 1. Music sequencers have a pre-recorded library of sounds (sound components) which are combined together to create music signal. Input to the sequencer is a MIDI file which contains information about beginning of music events in time, their duration, IDs of sounds (in our case the pre-recorded sound components), their amplitude and modification type. In this paper we consider only component truncation as a possible modification. Output of the sequencer is the audio signal. Input of our 'inverse music sequencer' is the recorded music signal and its output is the estimated (transcribed) MIDI-like representation of music events. Consideration of possible truncations of the events is a distinct feature of our approach in comparison to other approaches that consider only full sequences of the frames [3].

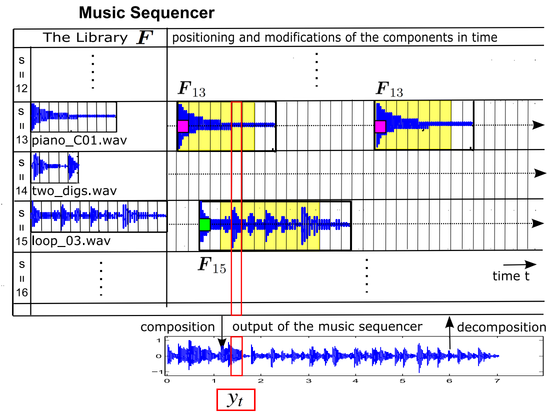


Figure 1: Principle of a music sequencer. The range of active frames is yellowed. Note that the amplitudes are the same for all events in a track s (represented by squares of the same color).

2. MATHEMATICAL MODEL

The recorded signal y_t is modelled by the following state space model:

$$y_t = \sum_{s \in S} a_s F_s l_{s,t} + e_t, \tag{1}$$

$$l_{s,t} = T l_{s,t-1}, \tag{2}$$

$$l_{s,t} = [0,0,\ldots,1,0\ldots,0]. \tag{3}$$

Here, y_t is the ϕ-dimensional vector of measurements at time t composed of either time- or frequency-representation of the input music signal segment (frame); the observations are corrupted with noise e_t of Gaussian distribution with zero mean and known covariance matrix $\omega^{-1} I_\phi$; $\mathscr{F} = \{F_1, F_2, \ldots F_S\}$ is a library of S pre-recorded sounds; the sound matrix $F_s = [f_{s,1}, f_{s,2}, \ldots, f_{s,l}]$ is a collection of temporal sequence of recorded segments for the isolated sound s. By convention, the first column $f_{s,1}$ is composed of zeros; the label process $l_{s,t} = [0,0,\ldots,1,0\ldots,0]$ denotes which frame of the sound is active at time t, $l_{s,t} = [1,0,\ldots]$ encodes that sound s is silent; a_s is the amplitude of the sound which is assumed to be constant in time[1]. The measurements y_t and the columns of F_s are represented by magnitudes of short time Fourier transform (STFT). The labels form an unobserved Markov

[1]This will be relaxed in Section 3.2.

model (2) with transition matrix:

$$T = \begin{bmatrix} t_{sil} & t_{end} & t_{end} & \dots & t_{end} \\ t_{start} & t_{stay} & & & \\ t_{start} & t_{next} & t_{stay} & & \\ \vdots & t_{skip} & t_{next} & t_{stay} & \\ t_{start} & & t_{skip} & t_{next} & t_{stay} \end{bmatrix}, \tag{4}$$

where t_{sil} denotes probability of the silent frame staying silent, t_{end} denotes probability of transition from a non-silent to the silent state, t_{start} denotes probability of transition from silence to non-silence, t_{stay} is the probability of repetition of the same frame, t_{next} is probability of continuation of the sound to the next frame, t_{skip} is probability that one frame in the sequence is skipped.

3. APPROXIMATE BAYESIAN IDENTIFICATION

The task is to estimate posterior probability of the hidden label process $L_t = [l_{1,t}, l_{2,t}, \dots l_{S,t}]$, i.e. labels of all sounds in the bank, and their corresponding amplitudes $a = [a_1, a_2, \dots, a_S]$. This can be formally achieved using the Bayes rule:

$$p(l_{1,1:t}, l_{2,1:t} \dots l_{S,1:t}, a | y_{1:t}) \propto \prod_{\tau=1}^{t} p(y_\tau | L_\tau, a) p(L_\tau | L_{\tau-1}) p(L_0) p(a). \tag{5}$$

$$p(L_\tau | L_{\tau-1}) = \prod_{s=1}^{S} p(l_{s,\tau} | l_{s,\tau-1})$$

where subscript $_{1:t}$ denotes a time sequence, e.g. $l_{s,1:t} = [l_{s,1}, l_{s,2}, \dots l_{s,t}]$. Prior distribution $p(L_0)$ is chosen as non-informative, in this case uniform. Prior distribution

$$p(a) = \mathcal{N}(\mu_{a,0}, \Sigma_{a,0}),$$

is chosen to regularize the model in case that a sound is not played at all. In such a case, the data $y_{1:t}$ are not informative about the amplitude of the sound and the posterior density is equal to the prior.

Exact Bayesian inference of model (1)–(3) via (19) is computationally intractable since the number of components in the likelihood (5) grows with time. Therefore, we propose to use approximate inference based on Variational Bayes approximation [4]. This technique was successfully used for on-line estimation of mixture models [5]. Following the methodology, we seek approximate inference only within conditionally independent posterior

$$p(l_{1,1:t}, \dots l_{S,1:t}, a | y_{1:t}) \equiv p(a | y_{1:t}) \prod_{s=1}^{S} p(l_{s,t} | y_{1:t}). \tag{6}$$

Minimizing Kullback Leibler divergence between the left- and the right-hand side of (6), we obtain the following set of implicit equations:

$$p(l_{s,1:t} | y_{1:t}) \propto \exp\left(\mathsf{E}_{a, l_\sigma, \sigma=1\dots S, \sigma \neq s}(\ln p(L_{1:t}, a, y_{1:t})\right) \tag{7}$$

$$p(a | y_{1:t}) \propto \exp\left(\mathsf{E}_{l_\sigma, \sigma=1\dots S,}(\ln p(L_{1:t}, a, y_{1:t}))\right) \tag{8}$$

Solution of this set is often found using an iterative algorithm [4].

Substituting (1) and (5) into (7) and necessary simplification, we obtain:

$$p(l_{s,1:t} | y_t) \propto \prod_{\tau=1}^{t} \prod_{i=1}^{dim(F_s)} o_{i,\tau}^{l_{s,i}=1} t_{i,:} l_{s,\tau-1} \tag{9}$$

$$o_{i,\tau} \propto \exp\left(-\frac{1}{2}\omega(\tilde{y}_\tau - \hat{a}_s f_{s,i})'(\tilde{y}_\tau - \hat{a}_s f_{s,i})\right)$$
$$\exp\left(-\frac{1}{2}\omega\Sigma_{a,s,s} f'_{s,i} f_{s,i}\right). \tag{10}$$

$$\tilde{y}_\tau = y_\tau - \sum_{\sigma=1, \sigma \neq s}^{S} \hat{a}_\sigma F_\sigma \hat{l}_{\sigma,\tau}, \tag{11}$$

$$p(a | y_{1:t}) = \mathcal{N}(\mu_a, \Sigma_a), \tag{12}$$

$$\mu_a = \Sigma_a \left(\sum_{\tau=1}^{t} \mathsf{E}(\Phi_\tau)' y_\tau + \Sigma_{a,0}^{-1} \mu_{a,0}\right), \tag{13}$$

$$\Sigma_a = \left(\sum_{\tau=1}^{t} \mathsf{E}(\Phi'_\tau \Phi_\tau) + \Sigma_{a,0}^{-1}\right)^{-1}. \tag{14}$$

The expectations are:

$$\mathsf{E}(\Phi'_\tau \Phi_\tau) = \omega[F_1 \hat{l}_{1,\tau}, \dots, F_S \hat{l}_{S,\tau}]'[F_1 \hat{l}_{1,\tau}, \dots, F_S \hat{l}_{S,\tau}], \tag{15}$$
$$\mathsf{E}(\Phi_\tau) = \omega[F_1 \hat{l}_{1,\tau}, \dots, F_S \hat{l}_{S,\tau}],$$

$\hat{l}_s$ denotes expected value of l_s, $\hat{a} = \mu_a$.

Note that (17) can be rewritten as

$$p(l_{s,1:t} | y_{1:t}) \propto \prod_{\tau=1}^{T} p(\tilde{y}_\tau | l_{s,t}) p(l_{s,t} | l_{s,t-1}) p(l_{s,0}). \tag{16}$$

which is a standard hidden Markov model that could be solved using the forward-backward algorithm. However, due to dependence of $\tilde{y}_\tau$ on the expected values $\hat{l}_\sigma$, it will be used only as a subroutine within the following algorithm:

Off-line VB algorithm

1. Set initial conditions, e.g. $\hat{l}_s^{(0)} = [1, 0, 0, \dots], \forall s$, and set iterative counter $i = 0$.
   (a) For $s = 1, \dots, S$
      i. compute $\tilde{y}_\tau$ using available estimates (11),
      ii. evaluate posterior $\hat{l}_s^{(i)}$ via forward-backward algorithm using (4) and (10).
      iii. update statistics of $p(a | y_{1:t})$ using (13)–(14).
      iv. $i = i + 1$.
   (b) If $i < max\_iterations$ and $distance(\hat{l}_s^{(i)}, \hat{l}_s^{(i-1)}) > threshold$ goto 2, end otherwise.

3.1 Extension for unknown precision of observations

The precision of observations ω was considered to be known in (1). When it is unknown, it can be estimated using the same methodology. In that case, we need to complement the likelihood (5) by a prior

$$p(\omega) = Ga(a_0, b_0),$$

and conditionally independent posterior of ω is then:

$$p(\omega|y_t) \propto Ga(a,b), \quad a = a_0 + \phi t, \tag{17}$$

$$b = b_0 + \sum_{\tau=1}^{t} (y_\tau - \sum_s \hat{a}_s F_s \hat{l}_{s,\tau})'(y_\tau - \sum_s \hat{a}_s F_s \hat{l}_{s,\tau}) + \sum_\tau \mathrm{trace}(\Sigma_a \mathsf{E}(\Phi_\tau)' \mathsf{E}(\Phi_\tau)) \tag{18}$$

and ω in (10) and (15) should be replaced by $\hat{\omega} = a/b$.

3.2 Extension of the algorithm to recursive estimation.

Algorithm 3 can be easily extended to recursive form by running Algorithm 1 on moving window of length w. In that case, we perform Bayesian estimation of the labels on the moving window, $L_{t-w:t}$, via the Bayes rule.

$$p(L_{t-w:t}|y_{1:t}) \propto \prod_{\tau=t-w}^{t} p(y_\tau|a_s, L_\tau) p(L_\tau|L_{\tau-1}) p(L_{t-w}) p(a). \tag{19}$$

The prior distribution $p(L_{t-w})$ in (19) is the delayed posterior. Note that amplitudes, a, are now considered stationary only with respect to the moving window.

4. EXPERIMENTS

The simulated data were generated from piano midi files in the same way as in [6]. Each note was represented by its pitch, onset time, duration and offset in the sound library. The last element is an extension of the standard midi. The amplitudes and the audio signal were generated using model (1-3). The sound library used to obtain the observed signal was identical to the library used in the estimation.

The sounds assigned to the piano midi events were synthesized according to $y_{acoust} = env \cdot \sin(2\pi f \cdot t + 5 \cdot env \cdot \sin(2\pi f \cdot t))$, where env denotes amplitude envelope and f represents the fundamental frequency, see in Fig. 3. Frames produced by such a synthesizer are significantly similar to each other. Hence, the audio signal generated by the first frame at low amplitude is remarkably similar e.g. to that of the third frame at higher amplitude. This is a challenge for estimation, since the model must be able to properly distinguish those two cases. The likelihood model makes at most one frame of a sound to be active in time t and the transition model concatenates frames corresponding to frame sequences from the library.

Elements of the transition matrix T form nuisance parameters of the model, $\delta = [t_{sil}, t_{start}, t_{end}, t_{stay}, t_{next}, t_{skip}]$. These were optimized by Matlab function fminsearch to optimize a measure similar to the total relative sound-to-distortion ratio [7]:

$$SDR = 10log_{10} \frac{\sum_t [b \cdot F_{acoust} a_t]^2}{\sum_t [y_t - b \cdot F_{acoust} a_t]^2}, \tag{20}$$

where b is a scalar fitting $b \cdot A = A_{reference} + noise$ according to MMSE and F_{acoust} is the matrix of frames in acoustic form.

A library of 61 sounds (corresponding to midi notes 36—96) was used for testing, each of the sounds having 10 frames. Each frame contained 4096 samples at 44.1 kHz sample rate, represented by the magnitude spectrum. For training of the nuisance parameters, only 36 (midi notes 45—80) sounds were considered. Thus, there were 610 and 360 frames in the testing and the training library, respectively. Model nuisance parameters were trained on 51 frame units long of one of Debussy's preludes and tested on 582 frame long concatenation of short excerpts of Mozart, Beethoven and Debussy. In the training phase, 51 units were filtered by the VB algorithm, frame by frame with no overlap. The SDR-optimized results are presented in Fig. 4. In the testing phase, 58 seconds of music audio signal containing 1325 active frames were estimated by the VB algorithm. In our experiments, five iterations of the VB algorithm were sufficient to achieve convergence. After five iterations we evaluated the fit of the presence vectors L_t using maximum aposteriori estimate, i.e. the frame with highest probability was declared as detected. The number of correctly detected frames, false positives has been decreased by the current model while the number of false negatives has been slightly increased, see Table 1. This is due to missed frames in the end of the long sounds, the frames correspond to weak sound since the amplitude envelope of a component decreases in time, see in Fig. 3. Visual comparison of the presented model with continuous model of [6] and NMF approach of [8] is presented in Fig. 4. Since NMF si computationally cheaper, we have used it as initializer of the proposed VB algorithm. Specifically, the initial estimate of L_t in step 1. of Algorithm 1 were set to the NMF result. From the figures and the result table it follows that the observation part of the model enhances the quality of estimation from the NMF initialization and that the transitional part of the model improves model estimation. For illustration, result of the amplitude estimation for each sound component can be seen in Fig. 4.

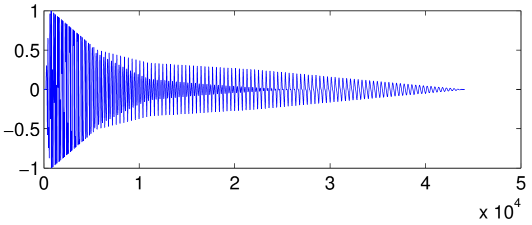


Figure 3: Amplitude envelope of sound 10, i.e., tone A (110 Hz).

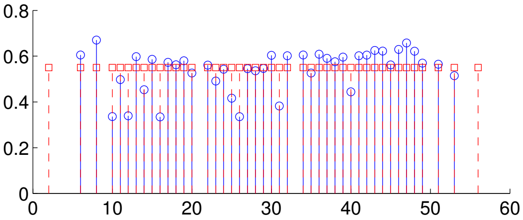
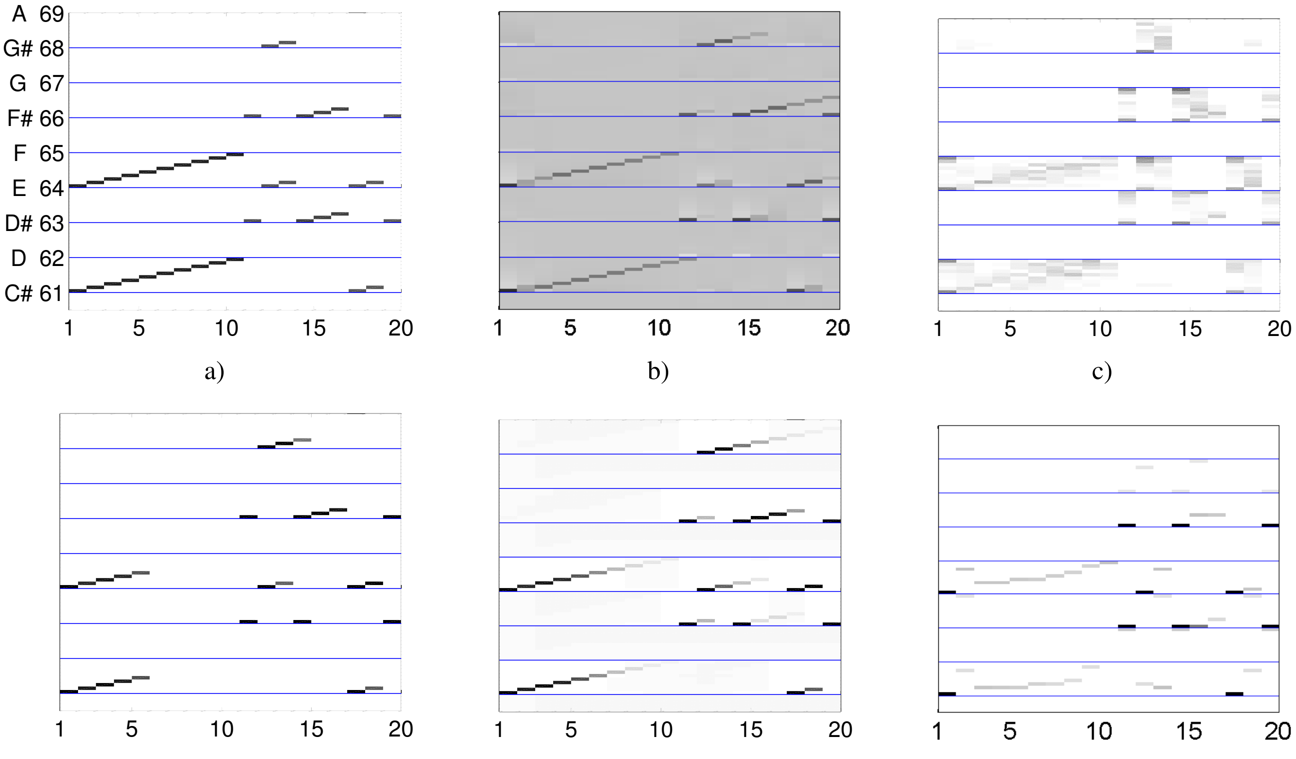


Figure 4: Amplitude estimation. Squares denote true values, circles represent estimated values. Missing squares / circles correspond to a state when the sound component is not present.

5. CONCLUSION

We have presented a model for polyphonic music signal and an approximate algorithm for estimation of its parameters based on Variational Bayes approximation. The experiments



a) b) c)

d) e) f)

Figure 2: Example of simulated and transcribed piece of polyphonic music in a piano-roll like representation. Vertical axes denote tone with the due midi keys. Horizontal axis denote discrete time (time units). Blue horizontal lines correspond to the beginning of the frame sequence due to a tone. **a)** original music excerpt; **b)** transcription via model[6]; **c)** transcription via NMF without any constraints; **d)** maximum likelihood estimates of the current frame using the current model; **e)** posterior values $\hat{L}$ of the current model; **f)** maximum likelihood estimates of the current frame using the current model without the transition part.

Table 1: Comparison of the presented model with previous methods.

	total frames	hits	false positive	false neg	SDR [dB]
current posterior	1325	1167	154	158	10.08
current observation	1325	1004	2908	321	5.19
[6]	1325	1219	293	106	10.59
NMF	1325	1007	1367	218	3.53

confirm that the off-line version of the algorithm compares favourably with competing algorithms. Extensive tests of the recursive version are under development. Further work will be directed to optimization of the implied computational load. The algorithm can be further extended to on-line estimation of the transition matrix of the label process.

REFERENCES

[1] M. Davy and A. Klapuri, eds., *Signal Processing Methods For Music Transcription*. Springer, 2006.

[2] K. Kashino and H. Tanaka, “A sound source separation system with the ability of automatic tone modeling,” in *International Computer Music Conference (ICMC)*, Aug. 1993.

[3] A. Mesaros, T. Heittola, A. Eronen, and T. Virtanen, “Acoustic event detection in real life recordings,” in *European Signal Processing Conference*, (Aalborg, North Denmark), 2010.

[4] V. Šmídl and A. Quinn, *The Variational Bayes Method in Signal Processing*. Springer, 2005.

[5] M. Sato, “Online model selection based on the variational Bayes,” *Neural Computation*, vol. 13, pp. 1649–1681, 2001.

[6] Š. Albrecht and V. Šmídl, “Improvements of continuous model for memory-based automatic music transcription,” in *European Signal Processing Conference*, (Aalborg, North Denmark), 2010.

[7] R. Gribonval, L. Benaroya, E. Vincent, and C. Févotte, “Proposals for performance measurement in source separation,” in *Proc. 4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, (Nara, Japan), pp. 763–768, 2003.

[8] D. D. Lee and H. S. Seung, “Algorithms for non-negative matrix factorization,” in *NIPS*, pp. 556–562, 2000.