EURASIP Journal on
Audio, Speech, and Music Processing
a SpringerOpen Journal

RESEARCH Open Access

Transcribing Bach chorales: Limitations and potentials of non-negative matrix factorisation

Somnuk Phon-Amnuaisuk

Abstract

This article discusses our research on polyphonic music transcription using *non-negative matrix factorisation (NMF)*. The application of NMF in polyphonic transcription offers an alternative approach in which observed frequency spectra from polyphonic audio could be seen as an aggregation of spectra from monophonic components. However, it is not easy to find accurate aggregations using a standard NMF procedure since there are many ways to satisfy the factoring of $V \approx WH$. Three limitations associated with the application of standard NMF to factor frequency spectra are (i) *the permutation of transcription output*; (ii) *the unknown factoring r*; and (iii) *the factoring W and H that have a tendency to be trapped in a sub-optimal solution*. This work explores the uses of the heuristics that exploit the harmonic information of each pitch to tackle these limitations. In our implementation, this harmonic information is learned from the training data consisting of the pitches from a desired instrument, while the unknown effective *r* is approximated from the correlation between the input signal and the training data. This approach offers an effective exploitation of the domain knowledge. The empirical results show that the proposed approach could significantly improve the accuracy of the transcription output as compared to the standard NMF approach.

Keywords: polyphonic music transcription, non-negative matrix factorisation, tone-models, transcribing Bach chorales

1 Introduction

Automatic music transcription concerns the translation of music sounds into written manuscripts in standard music notations. Important components for automated transcription are pitch identification, onset-offset time identification and dynamics identification. Research activities in this area have been reported in [1-19]. Up to now, it is still not possible to accurately transcribe polyphonic notes from an orchestra, a popular band or even a solo instrument. The mixture of sounds from different pitches pose difficulties for the existing techniques. To date, the transcription of a single melody line (monophonic) is quite accurate but transcribing polyphonic audio is still an open research area.

Commonly employed features in audio analysis could be derived from time domain and frequency domain components of the input sound wave. Transcribing a single melody line (i.e., monophonic case) involves tracking only a single note at any given time. The fundamental frequency, F_0, can usually be reliably estimated using autocorrelation in the time domain or by tracking the F_0 in the frequency domain. In the polyphonic case, multiple F_0 tracking has been attempted using both time domain and frequency domain approaches [20]. However, harmonic interference from simultaneous notes complicate the multiple F_0 tracking process. Standard techniques relying on either time domain or frequency domain approaches do not seem to be powerful enough to address the issue of harmonic interference.

This challenge has been approached from different perspectives, one of which is the blackboard architecture that incorporates various knowledge sources in the system [21]. These knowledge sources provide information regarding notes, intervals, chords, etc., which could be used in the transcription process. Explicitly encoded knowledge in this style is usually effective but requires a laborious knowledge engineering effort. Soft computing techniques such as the Bayesian approach [4,8,11,19,22] graphical modeling [23]; artificial neural networks [24];

Correspondence: somnuk@utar.edu.my
Music Informatics Research Group, Universiti Tunku Abdul Rahman, Selangor Darul Ehsan, Malaysia

and factoring techniques (e.g., ICA, NMF) [16,25] have emerged as other popular alternatives since knowledge elicitation and maintenance could be performed from the training data.

This article investigates the application of NMF for an automatic transcription task. Although this is not the first time for NMF to be applied in polyphonic transcription, this study is different because it addresses three limitations of the conventional automatic transcription using NMF (see [16]): (i) the permutation of transcribed notes; (ii) the determination of the factor r which plays a major role in the accuracy of the transcribed output; and (iii) the factorisation process via alternating projected gradient method that may get trapped in local optima.

These three issues will be addressed by the use of heuristics. In brief, polyphonic audio is transformed into its frequency domain counterpart as a matrix $V^{m\times n}$, where each column corresponds to the frequency m at time n. NMF factors the matrix V to two components $V^{m\times n} \approx W^{m\times r}H^{r\times n}$. In our approach, the columns of the matrix W contain r Tone-models that represent the frequency spectra of notes. Each row of matrix H is the weight corresponding to the activation of note r (i.e., the transcribed notes).

The scope of this article is limited to the discussion of polyphonic transcription of Bach chorales using NMF. The materials in this article are organised as follows: in Section 2, related studies are reviewed; in Section 3, the concepts behind our approach are discussed; in Section 4. The experimental results are presented and critically discussed; and finally Section 5 contains the conclusion of this study.

2 Related works

The transcription of polyphonic audio has a long history. Moorer [14] was among the pioneers who investigated automatic transcriptions from polyphonic audio. In his Ph.D thesis in 1975, he demonstrated the transcriptions of a two-part guitar duet as well as a synthesised violin duet (both examples have at most two notes being played simultaneously at any time). Moorer approached this problem by devising a comb filter for each musical note. Each comb filter had many narrow bandpass centered at all the harmonics of the note. The transcribed notes were inferred from the output of these comb filters.

There have been many variations to the research activities in transcribing polyphonic audio in the past few decades. Attempts to solve the polyphonic transcription problem could be viewed along a spectrum in which at one end is a knowledge-based approach and at the other end, a soft computing approach. Examples of a knowledge-based approach are the organised processing toward intelligence music scene analysis (OPTIMA) [11]; and the blackboard architecture [2,21]. A knowledge-based approach exploits relevant knowledge in terms of rules to assist decision-making process. For example, the blackboard architecture [21] houses thirteen knowledge sources which hierarchically deal with notes, intervals, chords, etc. Exploiting expert knowledge in problem solving is usually effective since specialised knowledge is explicitly coded for the task. However, there are well known bottlenecks in knowledge acquisition and knowledge exploitation in a conventional knowledge-based system, especially if the knowledge is encoded in terms of production rules. The bigger the knowledge-based system, the longer the decision process takes. A soft computing approach is more flexible in terms of knowledge acquisition and knowledge exploitation since knowledge can be learned from examples. Once the system has learned that piece of knowledge, the exploitation is very effective since the decision process does not involve traditional searches as in conventional knowledge-based systems.

Marolt [24] experimented with various types of neural networks (e.g., time-delay neural network, Elman's neural network, multilayer perceptrons, etc.) in note classification tasks. Seventy-six neural network modules were used to recognise 76 notes from A1 to C8. Each neural network was trained to recognise one piano note with the frequency spectral features from approximately 30,000 samples where one-third of them were positive examples. Soft computing approaches such as connectionism, support vector machine, hidden Markov model [23,24,26], etc., usually require complete training data as the performance of the model highly depends on the decision boundary constructed using the information from the training examples. Sometimes, this is an undesirable requirement. The Bayesian approach is one of the most popular techniques for polyphonic transcription tasks. This may be because it provides a middle ground between the effectiveness of encoding prior knowledge in the model (as in knowledge-based approaches) and the ability to cope with uncertainties (found in soft computing approaches). Bayesian harmonic models have been used in pitch tracking in [8,19]. A Bayesian model exploits the prior knowledge of fundamental frequency and the harmonic characteristics of notes produced by an instrument.

More recently, the non-negative factoring technique has received a lot of attention [16,27,28]. NMF factors a positive matrix V into two other positive matrices WH where W and H could bear the interpretation of additive parts of V. NMF has been used in many domains as a technique for part-based representation such as image recognition [28]. Smaragdis and Brown [16] were among the pioneers who exploited NMF in music transcription problems. They showed that NMF could be used to

separate notes from polyphonic audio. In a recent study by [29], a nearest subspace search technique is employed to find the weight factor (contribution) of different sources in a dictionary.

In [1], the dictionary of atomic spectra was learned from audio examples. The learned dictionary comprised atomic spectra, which could be mapped back to pitches. This learned dictionary represents the basis vector, which could be used to factor out the transcribed notes. It should be noted that the learned atomic spectra often could not successfully represent the spectral characteristic of each pitch. From the learning process, a note may be represented by more than one atomic spectra. Furthermore, the mapping process between the pitches and the atomic spectra must still be done manually. In our approach, the matrix W of basis vectors is learned from each pitch from a desired instrument. This ensures that the basis vector (a.k.a. dictionary, Tone-model) represents the harmonic structure of each pitch at the expense of the basis vector matrix being applicable for that particular instrument only (e.g., the Tone-model learned from a piano will not work well with, for example, a violin). Many applications such as a performance analysis module in a guitar tutoring system, could benefit from this.

3 Exploring NMF for polyphonic transcription

We investigate the application of NMF to extract polyphonic notes from a given polyphonic audio. Our research problem can be summarised and illustrated using Figure 1. Let S be unobserved MIDI note-on/off signals that produce audio signal $y(t)$. The source frequency spectra V derived from polyphonic audio could be seen as an aggregation of the components from the basis vector matrix W and their activation pattern H.

Intuitively, H should approximate the activation of note events if W could successfully learn the harmonic structure of those notes events. Although learning W from the data is flexible and adaptive, there is no known means to control or to guide the learning of W. If the basis vectors $\mathbf{w}_r$ in the matrix W do not successfully represent the basis of each note event, then this would result in an erroneous note transcription in the matrix H.

Conventionally, the initial values of W and H are randomly initialised and the NMF algorithms use alternating minimisation of a cost function to find the optimal values of H and W. In one step, W is fixed and H is updated, while in the next step, H is fixed and W is updated. This method often results in an erroneous transcribed matrix, H, since there are many plausible solutions that could satisfy $V \approx WH$. As pointed out in [30], it is impossible to separate polyphonic notes from a single polyphonic sound channel without employing some kind of constraints to the signal.

Here, we propose a novel strategy by constructing a basis vector matrix W using a Tone-model of the desired instrument (instead of randomly initialising W as in the standard NMF). Constraining W using Tone-models has many positive side effects. It resolves the issue of the permutation of transcribed output notes since the output notes would be in the same order as the employed Tone-models. Furthermore, we propose to employ heuristics to switch off the components corresponding to the inactive Tone-models (see Section 3.3). This should help improve the quality of the obtained solution, since the search is started with a more or less correct value of W.

3.1 Non-negative matrix factorisation

NMF decomposes the input matrix V into its basis vector matrix W and its activation matrix H as follows:

$$V \approx WH \tag{1}$$

where $W \in \mathcal{R}^{m \times r}$, $H \in \mathcal{R}^{r \times n}$, and all the elements in V, W and H are constrained to real positive numbers $V \geq 0$, $W \geq 0$, $H \geq 0$. We also assume that $r \leq m < n$. Lee and Seung [28,31], suggested two styles of cost function. One is to minimise the squared Frobenius norm $D_{\mathrm{F}}(V \,||\, WH)$ and the other is to minimise the generalised Kullback-Leibler (KL) divergence $D_{\mathrm{KL}}(V \,||\, WH)$ (see Equations 2 and 3). They also proposed multiplicative update rules, which compromised between speed and ease of implementation (of conjugate gradient

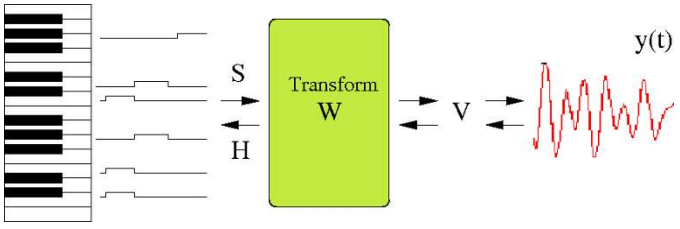

Figure 1 Problem statement: Determining notes from polyphonic audio could be seen as solving for the unobserved MIDI signal *S* from observable audio signal *y(t)*. If W characterises Tone-model components, then the unobserved MIDI could be estimated from V and W.

and gradient descent). Note that V_{mn} denotes an entry at row m and column n of the matrix V.

$$D_{\mathrm{F}}(V||WH) = ||V - WH||^2 = \sum_{mn} (V_{mn} - (WH)_{mn})^2 \quad (2)$$

$$D_{\mathrm{KL}}(V||WH) = \sum_{mn} (V_{mn} \log \frac{V_{mn}}{(WH)_{mn}} - V_{mn} + (WH)_{mn}) \quad (3)$$

3.2 Knowledge representation

Let $\mathbf{x}$ be a vector representing a sequence x_n, where x_1 is sample number one, sampled from analog audio signal with a sampling rate f_s. The sequence of discrete input samples x_n could be transformed from its time domain representation to its frequency domain counterpart using Fourier transform. A discrete Fourier coefficient X_k is defined as follows:

$$X_k = \sum_{n=0}^{N-1} h_n x_n e^{-j\frac{2kn\pi}{N}} \quad (4)$$

where N is the number of samples in a single window; h_n is the hamming window defined as $0.54 - 0.46\cos(2\pi\frac{n}{N})$; x_n are the time domain samples; k is the coefficient index and X_k is the corresponding frequency domain component. Each X_k coefficient is a complex number; its corresponding magnitude and phase represent the corresponding magnitude and phase of frequency at $k\frac{f_s}{N}$ Hz, where $k = 0, \ldots, \frac{N}{2}$.

3.2.1 Piano roll representation

It is decided that the input to NMF be abstracted at the activation level of each pitch in the standard equal tempered scale [32]. This abstraction reduces the size of the input vector significantly (as compared to using the magnitude of STFT coefficients). Smaller input size also reduces the computation effort required for the same task. In our representation, the input matrix V and the matrix W are represented as piano rolls, where the center frequency of each pitch i in the piano roll is calculated using the following equation:

$$f_c(i) = 440 \times 2^{(i-69)/12} \quad (5)$$

where i is the MIDI note number, i.e., 60 denotes middle ĉ, C4 (note that C3 is pitch ĉ that is an octave below C4 and C5 is pitch ĉ that is an octave above C4). The magnitude of the pitch i is the average of the magnitude of FT coefficients in the range of $0.99f_c(i)$ to $1.01f_c(i)$. For example, according to Equation 5, the pitch C4 has the center frequency of 261.63 Hz and has the lower and upper boundaries of 259.0 Hz and 264.2 Hz, respectively. With a sample rate f_s = 44, 100 Hz and window size N = 8192, these correspond to $k \in \{48, 49, 50\}$.[a] Hence, the activation magnitude of the pitch i can be calculated using the equation below:

$$\mathrm{pitch}(i) = \frac{1}{k_u - k_l + 1} \sum_{k=k_l}^{k_u} ||X_k|| \quad (6)$$

where k_l and k_u is the lowermost and the uppermost k index for the pitch i. The sequence of values of pitch(i) form a column of a piano roll.

3.2.2 Representing input V

At each time step, a short time fourier transform (STFT) is employed to transform the input sound wave into its frequency counterpart. Here, the STFT window is set to 8192 samples. The frequency resolution between each fourier transform (FT) coefficient is 5.38 Hz. These FT coefficients are binned according to the pitch on a piano (see Equation 6). For example, the input $\mathbf{v}_n$ of a monophonic note C4 would show the overtone series of pitch C4. The input V is presented in a piano roll representation by concatenating the column vectors $\mathbf{v}_n$ to form the matrix $V = [\mathbf{v}_1 \ldots \mathbf{v}_n]$.

3.2.3 Representing the tone-model

In our implementation, the basis vector matrix (Tone-model), W_{tm}, is also represented in a piano roll representation. The matrix W_{tm} is called the Tone-model, since it describes the harmonic structure of the pitches of an instrument. The matrix W_{tm} is calculated from a set of training examples which are monophonic pitches from C2 to B6. The magnitude of FT coefficients obtained from each training pitch are averaged across time frames and then binned to each pitch on the piano roll using Equation 6. Hence, each column of W_{tm} represents the Tone-model of each pitch. The matrix W_{tm} is constructed by concatenating the column vectors $\mathbf{w}_r$ together to form $W_{tm} = [\mathbf{w}_{c2} \ldots \mathbf{w}_{b6}]$.

3.3 Proposed transcription strategy

The overall concepts are outlined and summarised below. According to Figure 2, the Tone-models W_{tm} are learned in a separate offline process. At run time, polyphonic audio is transformed into the input matrix V. At each time frame, the correlation between V and each component in W_{tm} is computed and this information is employed as a heuristic to guess which $\mathbf{w}_r$ components in the Tone-models should be switched-off. The NMF process initialises the matrix H with random values uniformly distributed on the closed interval 0[1]. NMF updates H and W until WH successfully approximate V, i.e., V - $WH \leq$ acceptable error. The switching-off heuristics and the NMF procedure implemented in this work will be discussed next.

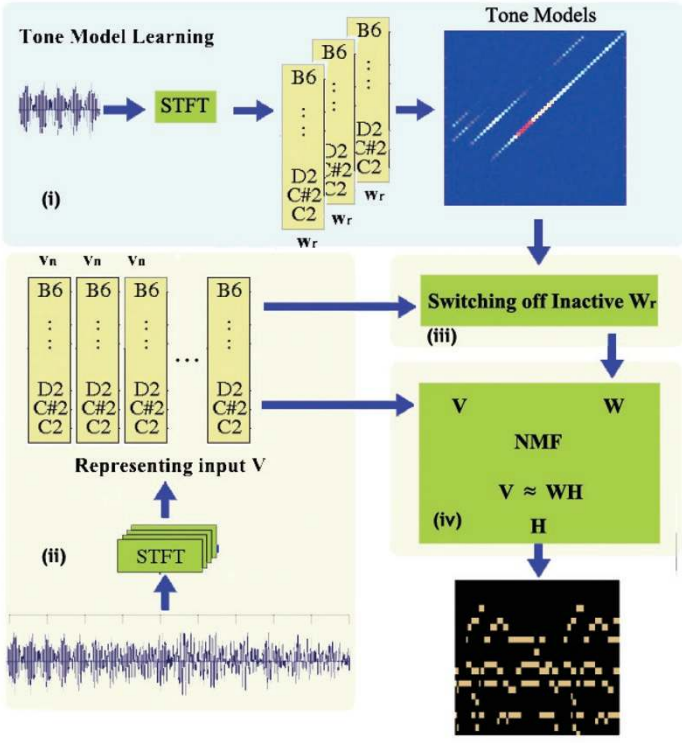

Figure 2 An overview of our approach: (i) Off-line Tone-model learning, **(ii)** Fourier transform and piano roll conversion, **(iii)** Switch-off inactive pitches and **(iv)** NMF process.

3.3.1 Switching off inactive pitches

Probable active pitches are guessed by comparing input V with the Tone-model W_{tm}. This is based on the fact that since W_{tm} is a matrix representing Tone-model vectors, each column $\mathbf{w}_r$ represents the Tone-model of each pitch from C2 to B6. For a given time frame n, if there is no sounding note, then entries of $\mathbf{v}_n$ are expected to be less than the entries of the Tone-model vectors $\mathbf{w}_r$. On the other hand, entries of $\mathbf{v}_n$ are expected to share common harmonic structures with $\mathbf{w}_r$ if the note r is sounding. Hence, an overlap in an overtone series between the input $\mathbf{v}_n$ and the Tone-model $\mathbf{w}_r$ is defined as a vector OL_r:

$$OL_r = \max(\mathbf{w}_r - \mathbf{v}_n, 0) \tag{7}$$

The ratio $||OL_r||/||\mathbf{w}_r||$ has its value lie in the closed interval 0[1]. OL_r is 1 when there is no overlap and OL_r is 0 when $\mathbf{w}_r$ is completely overlapped by $\mathbf{v}_n$. The note r is considered not sounding if the ratio $||OL_r||/||\mathbf{w}_r||$ is more than a threshold value and considered sounding if it is otherwise. The threshold value is empirically determined.

This heuristic is used to guess whether the pitch r is active by comparing the input spectrum at a time frame n to all the $\mathbf{w}_r$ and flagging the active pitch r. For each time frame n, a vector $L_n = [l_1, \ldots, l_r]^T$ estimates whether the pitch r is active or inactive. After running through all the time frames of the input signal, the active pitches are determined as a disjunction of all the active pitch flags $L = L_1 \vee L_2 \vee \ldots \vee L_n$. The pseudo code below summarises this process.

function *probablePitch*(W_{tm}, V) **return** $L^{r\times 1}$ an active pitch vector

```
for each v_n associated with time frame n
    L_n = [ ]
    for each w_r of each Tone-model r = 1, 2 , . . . ,
    60
        if ||OL_r||/||w_r|| >threshold
        then l_r = 1 else l_r = 0
        end
        L_n ← append(l_r,L_n)
    end
end
L ← L_1 ∨ L_2 ∨ . . . ∨ L_n
return L

end
```

The switch L estimated from the input V is used to switch off irrelevant basis vectors $\mathbf{w}_r$, i.e., the

constrained $W = W\text{diag}(L)$, where diag(L) returns a diagonal matrix.

3.3.2 Transcribing polyphonic notes using NMF

Multiplicative update rules that minimise the cost functions (see Equations 2 and 3) proposed by Lee and Seung [31] are considered to be the standard NMF algorithms [33]. They are guaranteed to find at least locally optimum solutions [31]. The update rules in (Equations 8 and 9) corresponding to $D_{\text{KL}}(V \,||\, WH)$ are reproduced here for readers' convenience.

$$H_{rn} \leftarrow H_{rn} \frac{\sum_m W_{mr} V_{mn} / (WH)_{mn}}{\sum_{m'} W_{m'r}} \tag{8}$$

$$W_{mr} \leftarrow W_{mr} \frac{\sum_n H_{rn} V_{mn} / (WH)_{mn}}{\sum_{n'} H_{rn'}} \tag{9}$$

Factoring the matrix V to two components W and H can be viewed as a search process. The update rules guide the search to a solution using the gradient of the cost functions. There are many plausible solutions that could satisfy this factoring. Sometimes, the search will get stuck in sub-optimum solutions. Most search techniques could benefit from extra knowledge introduced in terms of constraints. Depending on applications, extra information introduced to guide the search can be in different forms. In our application, initialising the basis vector matrix W with Tone-models and switching off inactive r components help initialise the search near optimum solutions and better solutions are usually obtained if the search starts near good solutions in the search space. In our implementation, the cost function updated is closely related to *expectation maximisation maximum likelihood (EMML)* which has been studied in image processing [34]. In EMML, H is iteratively updated while W is assumed to be known and fixed. In our experiment, H is updated using (following [28]):

$$H_{rn} \leftarrow H_{rn} \sum_m W_{mr} \frac{V_{mn}}{(WH)_{mn}} \tag{10}$$

In our experiment, two experimental designs have been carried out: (i) Tone-model NMF (TM-NMF) where the matrix W is initialised using the Tone-model W_{tm} and r constraint and its values are fixed throughout the run; and (ii) initialised constrained Tone-model (ICTM-NMF), where W is initialised in the same fashion as in TM-NMF but W is updated as below:

$$W_{mr} \leftarrow W_{mr} \frac{\sum_n H_{rn}}{\max_r \left(\sum_n H_{rn}\right)} \tag{11}$$

$$W_{mr} \leftarrow \frac{W_{mr}}{\sum_m W_{mr}} \tag{12}$$

A column of V is formed from columns of W weighted by value given in H. In other words, a column of H is a new representation of a column of V based on the basis of W. Hence, each $\mathbf{w}_r$ is updated by scaling it to the predicted activation of $\max_r(\Sigma_n H_{rn})$ (Equation 11), each $\mathbf{w}_r$ is then normalised (Equation 12). The pseudo code below summarises the two NMF processes (TM-NMF and ICTM-NMF) employed in our experiments.

function *transcribeBach*(W_{tm}, V) **return** Pitch activation H

$L \leftarrow probablePitch(W_{tm}, V)$
Initialise H randomly s.t. $H_{rn} \in \{h | 0 \le h \le 1\}$
Initialise W using Tone-models and heuristics; $W \leftarrow W_{tm} diag(L)$
/* Stopping criteria:
(i) Exceed max-iteration-set at 3000 iterations, or
(ii) The matrix H converges, their values become stable, or
(iii) V - $WH \le$ acceptable error */
while some stopping criteria is not satisfied
update H using Equation 10
if TM-NMF **then** $W \leftarrow W_{tm} diag(L)$
if ICTM-NMF **then** update W using Equations 11, 12
end
return H

end

The output H is then converted to a binary (note on/off) by applying a threshold to it. To evaluate H, the note on/off information of each original chorale is extracted from the MIDI file. This forms a ground truth for each chorale. In this process, the MIDI time is retimed to linearly map with the number of frames in H.

4 Experimental results

In this experiment, the input wave files were generated by playing back MIDI files using a standard PC sound card. Recording was done with 16 bit mono and with a sampling rate of 44100 Hz. The recorded wave file was transformed to the frequency domain using STFT. The STFT window size was set at 8192 samples. The Hamming window function was applied to the signal before converting it to the frequency domain. The experimental results of Bach chorales are summarised in Table 1. Two variations of the Tone-model usage were carried out (i) TM-NMF, and (ii) ICTM-NMF.

4.1 Evaluation measures

The literature uses a variety of ways to define the correct transcription of notes. Should a note be classified as correctly transcribed or incorrectly transcribed if the note is accurately transcribed in terms of pitch but the

Table 1 Summary of performance of TM-NMF, and ICTM-NMF in transcribing Bach chorales.

ID		TM-NMF			ICTM-NMF		
		Prec	Recall	F	Prec	Recall	F
10	Aus tiefer Not schrei ich zu dir	0.54	0.55	0.55	0.63	0.63	0.63
26	O Ewigkeit, du Donnerwort	0.65	0.62	0.63	0.67	0.78	0.72
28	Nun komm, der Heiden Heiland	0.64	0.59	0.61	0.66	0.70	0.68
48	Ach wie nichtig, ach wie flüchtig	0.74	0.57	0.64	0.60	0.78	0.67
100	Herr Christ, der ein'ge Gott's-Sohn	0.54	0.55	0.54	0.62	0.64	0.63
102	Ermuntre dich, mein schwacher Geist	0.63	0.62	0.62	0.69	0.70	0.70
156	Ach Gott, wie manches Herzeleid	0.72	0.54	0.62	0.65	0.76	0.70
182	Wär' Gott nicht mit uns diese Zeit	0.59	0.56	0.57	0.66	0.69	0.67
266	Herr Jesu Christ, du höchstes Gut	0.56	0.61	0.58	0.65	0.69	0.67
279	Ach Gott und Herr	0.66	0.59	0.62	0.67	0.68	0.68
290	Es ist das Heil uns kommen her	0.70	0.58	0.63	0.67	0.71	0.69
305	Wie schön leuehtet der Morgenstern	0.61	0.59	0.59	0.66	0.71	0.68
321	Wir Christenleut'	0.69	0.54	0.60	0.60	0.81	0.69
355	Nun ruhen alle Wälder	0.62	0.59	0.60	0.55	0.70	0.62

duration is not exact? In [35], note detections were calculated on each frame. The transcription output was converted to a binary note on/off and was compared to MIDI note on/off on a frame by frame basis. This was a good approach since it took the note duration into account. This work evaluated the transcribed output using the same approach in [35]. The results were evaluated based on the standard precision and recall measures where, in each frame, true positive *tp* is the number of correctly transcribed note events, false positive *fp* is the number of spurious note events and false negative *fn* is the number of note events that are undetected.

The true positive was calculated based on the matched pixels between the original piano roll and the transcribed piano roll H (i.e., *Original* and *Transcribed* in Equations 13, 14 and 15). False positive and false negative were calculated from the unmatched pixels.

$$tp = \max((\text{Original} + \text{Transcribed}) - 1, 0) \tag{13}$$

$$fp = \max(\text{Transcribed} - \text{Original}, 0) \text{ and} \tag{14}$$

$$fn = \max(\text{Original} - \text{Transcribed}, 0) \tag{15}$$

where max(a, b) returned the a if $a >= b$ otherwise returned b. The *Original* and *Transcribed* were $r \times n$ binary matrices (note on = 1 and note off = 0). The *Transcribed* matrix was obtained by thresholding the output H (see Section 3.3.2). The *Original* matrix was obtained by time-scaling the note on/off matrix to match the number of time frames in H. In this study, the note on/off matrix was obtained from the note on/off events extracted from the MIDI files and this provided the ground truth reference.

We resort to the *precision, recall* and *f* measures to judge the performance of the system. Precision provides measurement on the percentage of the correct transcribed note-on events from all the transcribed note-on events. Recall provides measurement on the percentage of the correct transcribed note-on events from all actual note-on events (i.e., reference ground truth).

In the transcription task, the precision and recall measures are equally important since it is undesirable to have a system with high precision but poor recall (or vise versa). Hence, the f-measure is computed since it provides an evenly weighted result of both precision and recall measures. These measures are defined as:

$$precision = \frac{tp}{tp + fp} \tag{16}$$

$$recall = \frac{tp}{tp + fn} \tag{17}$$

$$f = \frac{2 \times precision \times recall}{precision + recall} \tag{18}$$

4.2 Transcribing Bach chorales using ICTM-NMF

Figure 3 illustrates the effectiveness of our approach in tackling the weakness of applying standard NMF; the top pane shows the piano roll from the original chorale; the bottom left pane shows the output from a standard NMF where W and H are randomly initialised and r set at 60. Noise is observed in many places. This is a common problem with standard NMF algorithms used for transcribing pitches. This problem was mitigated with our proposed NMF with a constrained Tone-model. A great improvement in the output quality was observed in Figure 3 (bottom right pane). Although the transcribed output did not show an exact match to the input piano roll, a great improvement was observed in the bottom right pane.

Figure 4 shows the piano roll output from the transcription of chorale *Aus tiefer Not schrei ich zu dir* using ICTM-NMF. The first row is a piano roll representation of an original chorale. The second row shows

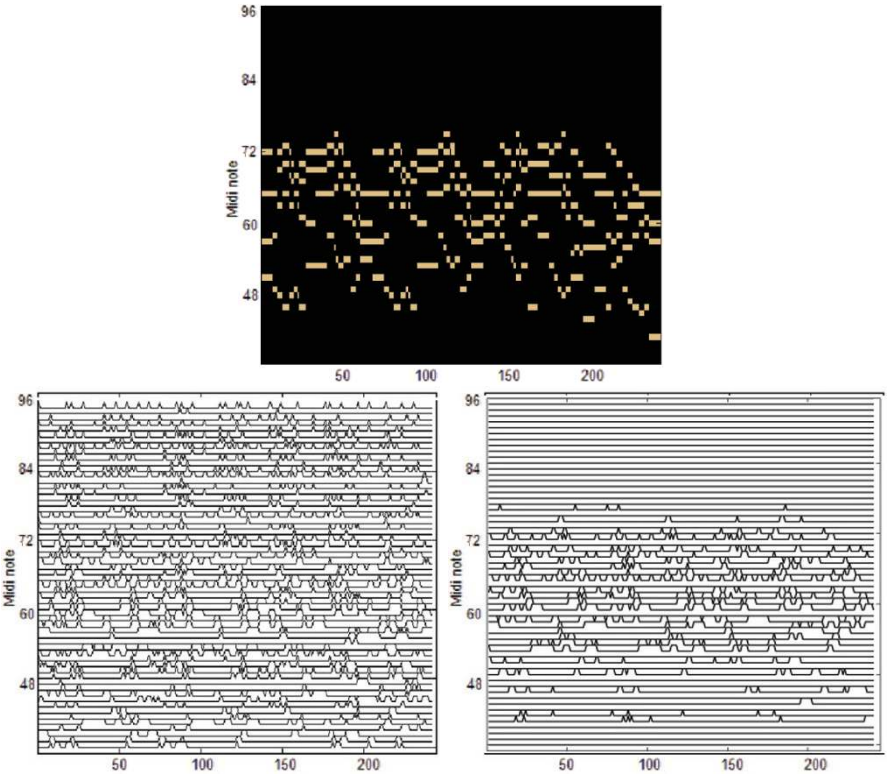

Figure 3 Effectiveness of ICTM-NMF as compared to standard NMF. Top pane: original piano roll of chorale *Aus tiefer Not schrei ich zu dir*; Bottom left pane: the transcription output from a standard NMF. Bottom right pane: the transcription output of the same chorale with ICTM-NMF. The y-axes represent the pitches from C2 (MIDI note number 36) to C7 (MIDI note number 96) while the x-axes represent time.

the transcribed output. The fourth and the fifth rows are *fp* and *fn*, respectively. From the Figure, *fp* and *fn* were calculated from the differences between the original chorales and the transcribed chorales.

Table 1 summarises the transcription results of Bach chorales. A total of fourteen chorales were arbitrarily chosen (chorales ID follows Riemenschneider. 371 harmonized chorales and 69 chorale melodies with figured bass). The input wave files of all the Bach Chorales used here were obtained by playing back the Bach chorale MIDI files downloaded from http://www.jsbchorales.net/bwv.shtml.

The output from ICTM-NMF shows a great improvement over the output from TM-NMF (around 7.5% improvement in *f* values). ICTM-NMF differed from TM-NMF in the following points: the Tone-models (i.e., the matrix W) was fixed in the TM-NMF but not fixed in ICTM-NMF. The W was allowed to be varied in ICTM-NMF, subjected to the constraint $\sum_m W_{mn} = 1$. As a consequence from the above point, all active basis vector (columns of W) remained active in TM-NMF. However, it was possible for active basis vectors in ICTM-NMF to be inactive during the W update process.

4.3 Performance comparison with related works

4.3.1 Beethoven's Bagatelle Opus 33, No. 1 in E

In this report, two transcriptions of the pieces demonstrated in previous studies were carried out using our proposed method. The first one was the transcription output from Beethoven's Bagatelle using NMF presented in [35]. The input sound wave, in [35], was recorded from a MIDI controlled acoustic piano.

The plot between recall and specificity[b] is reproduced in Figure 5 along with the output from our approach. Varying the threshold values that control the binary note on/off conversion of the output H produces the performance curve plot shown in Figure 5. The plot can be used to visually compare the performance of our system to the NMF output in [35]. The optimal *f* values for both NMF runs in the previous work were about 0.54 and 0.60 while our system obtained the optimal *f* value of 0.72 (recall 73.29% and precision 70.56%).

4.3.2 Mozart's piano Sonata No. 1 (KV279)

There are three movements in this sonata: Allegro, Andante and Allegro. Polyphonic transcription of the first two minutes of the first movement from KV279 was attempted using non-negative matrix division in [15]. Here, it was decided that, the whole first movement

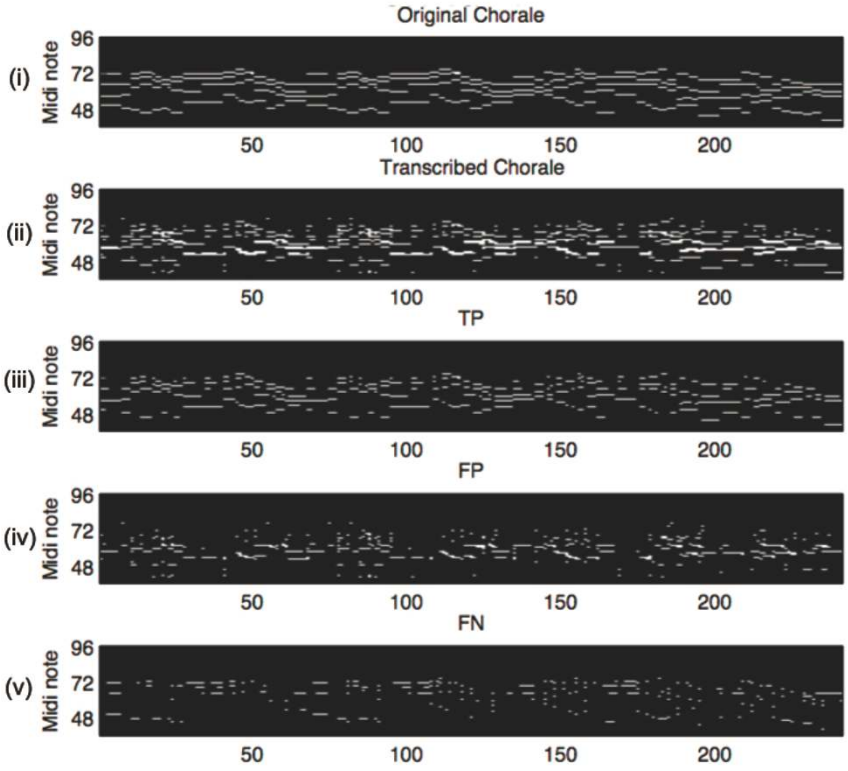

Figure 4 Experimental results (ICTM-NMF): from top to bottom **(i)** input piano roll of chorale *Aus tiefer Not schrei ich zu dir* **(ii)** transcribed output, **(iii)** true positives-*tp*, **(iv)** false positives-*fp*, and **(v)** false negative-*fn*.

would be used in our experiment. The main difference in our work is that in [15], the update of W step (see 3.1) was omitted. The input sound wave was recorded from a MIDI controlled synthesised piano in our experiment while the input sound wave was recorded from a computer controlled Bösendorfer SE290 grand piano in [15]. It was reported that the recall rate was 99.1%, the precision rate was 21.8% and the f value was 0.35. The issue of poor f value was tackled in [15] by further post-processing the output from NMF using classifiers (rule

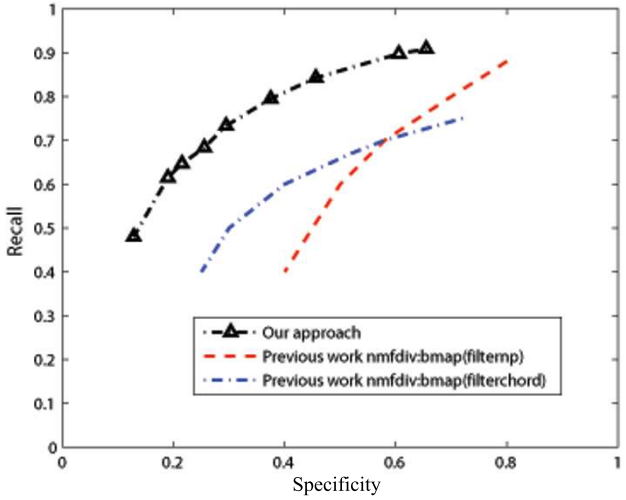

Figure 5 The plots between *Recall* and *Specificity* (i.e., *1-Precision*) for the transcription of Beethoven's Bagatelle. The optimal *f* value in our work is 0.72 while the optimal *f* were about 0.54 and 0.60 in the previous work.

based, instance based and frame based). This improved the *f* value significantly. Unfortunately, due to the limited length of [15], information given about the process was incomplete. There was no transcription output from [15], so a visual inspection of the output generated by both systems was not possible. For this piece, our approach yielded the optimal *f* value of 0.63 (recall 63.0% and precision 63.9%).

The performance statistics reported in our experiments were calculated using the precision and recall measures based on the graphical representation of a piano roll (as discussed in Section 4). It should also be pointed out that the counting of true-positive in [15] was based on correctly found notes,[c] which is unlike our true-positive which was based on frame by frame counting. The evaluations of the transcription of Beethoven's Bagatelle in [35] and our study have been based on similar assumptions.

4.4 Transcribing polyphonic sound from acoustic instruments

Sounds produced from real acoustic instruments possess a much more complex harmonic structure. The manner of note executions, the physical characteristic of the string, the soundboard, etc., all work together to determine the harmonic structures. The dictionary approach, such as the proposed Tone-model, represents complex harmonic structure of a note using a static Tone-model prototype. A static dictionary might not be effective in such a circumstance. Thus it is important to test the performance of the proposed approach on real acoustic musical instruments.

For this purpose, the chorale numbers 10, 26 and 28 were played on a classical acoustic guitar (model Yamaha CG 40) and on an upright acoustic piano (model Atlas). The sound was recorded directly via a single micropone with 16 bit bit-depth, and a sample rate of 44,100 Hz. The microphone had the following specifications: frequency response: 20 Hz - 16 KHz, sensitivity: -58 ± 3 dB, S/N ratio: 40 dB.

The transcription accuracy of polyphonic pieces performed by acoustic and synthesised instruments is displayed in Table 2. It was observed that the transcription accuracy obtained from acoustic sources was generally poorer than those from synthesised sources. Figure 6 shows the transcription output from the synthesised guitar sound and the acoustic guitar sound. The transcription output from synthesised sound (first row) shows better recall than the output from the acoustic sound (third row). It was observed from the experiment that transcription output from acoustic instruments tended to give inaccurate duration even though the pitch was correctly transcribed. This was common at a high pitch range region. The synthesised instruments did not suffer from this behaviour.

The overlays of the true positive output on the original chorale (the second and the fourth rows of Figure 6) shows that the degradation in performance in the acoustic case is mainly from the inaccuracy in transcribed duration. This could be caused by the harmonic complexity of real acoustic instruments and, from our observation, the faster decay rate of acoustic sound as compared to the synthesised sound (especially at the high pitch range).

We would also like to highlight that the degrading performance from the discrepancies in the duration did highlight the potential of our proposed approach. It implies that fine tuning in duration using information from the onset-offset time would greatly improve the quality of the transcriptions.

5 Conclusions

In this article, we proposed a new strategy to tackle the three limitations of standard NMF in the polyphonic transcription task. By constructing a basis vector matrix W using a Tone-model of the desired instrument and relying on heuristics to switch off the components corresponding to the inactive pitches, the experimental results showed an improvement in the transcription performance. This strategy worked because of the importance of the learned basis vector matrix and the ability of the NMF to switch off inactive basis vectors.

Table 2 Summary of performance of ICTM-NMF in transcribing Bach chorales from acoustic sound and synthesised acoustic sound

		Acoustic sound			Synthesised sound		
		Prec	Recall	F	Prec	Recall	F
	Instrument: **Guitar**						
10	Aus tiefer Not schrei ich zu dir	0.46	0.54	0.50	0.75	0.78	0.76
26	O Ewigkeit, du Donnerwort	0.39	0.57	0.46	0.72	0.74	0.73
28	Nun komm, der Heiden Heiland	0.37	0.55	0.45	0.71	0.75	0.73
	Instrument:**Piano**						
10	Aus tiefer Not schrei ich zu dir	0.58	0.53	0.56	0.63	0.63	0.63
26	O Ewigkeit, du Donnerwort	0.46	0.47	0.47	0.67	0.78	0.72
28	Nun komm, der Heiden Heiland	0.51	0.48	0.50	0.66	0.70	0.68

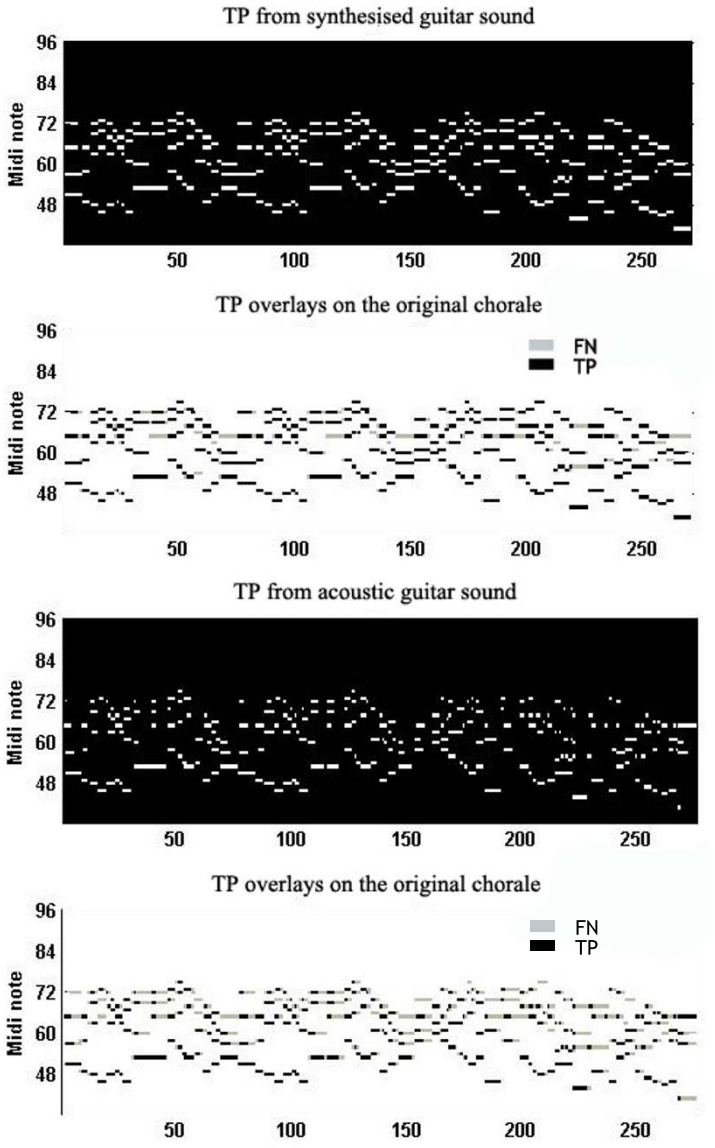

Figure 6 Plots of true positive obtained from syntheised guitar sound (first row) and from acoustic guitar sound (third row). The true positive values are overlaid on top of the original chorale: overlay of synthesised guitar sound (second row), overlay of acoustic guitar sound (fourth row). Note that in the overlaid *tp* on the original chorale, the colour code of the correct transcription (*tp*) is presented in white colour and the missing transcription (*fn*) is presented in gray colour.

The number of r played a crucial role in extracting note events. If the number of r was set higher than the actual active pitches, noise would appear as transcribed notes. On the contrary, if the number of r was set too low, events from different pitches would be transcribed as coming from the same pitch. To find the exact number of r is therefore a big challenge for polyphonic transcription using NMF [16]. In recent works by [6,15], NMF with a fixed W that learned from a desired instrument was proposed. In these works, the dictionary matrix, the pitch templates and the Tone-models acted as the basis vector matrix. This work extended the same concept to handle common limitations of NMF in polyphonic transcribing application.

Initialising the basis vector matrix using FFT spectra as a Tone-model is a powerful heuristic. However, this alone would not be enough to produce good output. From our study, the following heuristics should be included.

1. The Tone-model must characterise the input instrument;

2. the estimated r should be equal to or more than the actual r; and

3. the fixed Tone-model might not work well if r is not accurate.

To elaborate on the above heuristics, let us compare the NMF to a search process. If the NMF factoring process is seen as a search, the act of initialising W with a Tone-model is analogous to starting the search near the global optimum. When the search begins, fixing W biases the search to a certain direction. If the basis vector matrix W characterises the Tone-model of the input instrument and the value of active pitches r are determined correctly, then, it is likely that the obtained solution would be of good quality. If the value of r is wrongly determined, then the search might be guided to any non-optimal solution. Allowing the W to vary should lower the magnitude of inactive $\mathbf{w}_r$ and it is possible to compensate for an overestimated number of r. The experiment showed that the best results were obtained when the W was initialised using Tone-model and W was also allowed to be adjusted. In future work, we hope to further explore the extension of the Tone-model concept to handle sound produced from acoustic instruments.

Endnotes

[a]The index k might need to be rounded up/down since the boundary frequency of each pitch would not fall exactly on the desired value. [b]Specificity = 1-Precision. [c]As reported in [15]: "A note event is counted as correct if the transcribed and the real note do overlap".

Acknowledgements

We wish to thank anonymous reviewers for their comments, which help improve this article. We would also like to thank IPSR-Universiti Tunku Abdul Rahman for their partial financial support given to this research.

Competing interests

The author declares that they have no competing interests.

Received: 10 May 2011 Accepted: 27 February 2012
Published: 27 February 2012

References

1. SA Abdallah, MD Plumbley, Polyphonic music transcription by non-negative sparse coding of power spectra, in *Proceedings of International Conference on Music Information Retrieval (ISMIR 2004)*, Barcelona, Spain, 2004, pp. 318–325
2. JP Bello, Toward the automated analysis of simple polyphonic music: a knowledge-based approach, Ph.D. dissertation, Department of Electrical Engineering, Queen Mary, University of London, London, UK (2003)
3. GJ Brown, M Cooke, Perceptual grouping of musical sounds-a computational model. J New Music Res. **23**(2), 107–132 (1994). doi:10.1080/09298219408570651
4. AT Cemgil, B Kappen, D Barber, Generative model based polyphonic music transcription, in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, USA, 2003, pp. 181–184
5. C Chafe, D Jaffe, Source separation and note identification in polyphonic music, in *Proceedings of IEEE international Conference on Acoustic Speech and Signal Processing*, Tokyo, Japan, 1986, pp. 1289–1292
6. A Cont, Realtime multiple pitch observation using sparse non-negative constraints, in *Proceedings of the 7th International Symposium on Music Information Retrieval*, (ISMIR), Victoria, BC, Canada, 2006
7. RB Dannenberg, N Hu, Polyphonic audio matching for score following and intelligent audio editors, in *Proceedings of the International Computer Music Conference (ICMC 2003)*, San Francisco, USA, 2003, pp. 27–33
8. M Davy, SJ Godsill, Bayesian Harmonic Models for Musical Signal Analysis, in *Bayesian Statistics 7*, ed. by Bernardo JM, Bayarri MJ, Berger JO, Dawid AP, Heckerman D, Smith AFM, West M (Oxford University Press, Oxford, 2003), pp. 105–124
9. S Dixon, On the computer recognition of solo piano music, in *Proceedings of the Australian Computer Music Conference*, Brisbane, Australia, 2000, pp. 31–37
10. M Goto, A real-time music-scence-description system: predominant-F0 estimation for detecting melody and bass lines in real-world audio signals. Speech Commun. **43**, 311–329 (2004). doi:10.1016/j.specom.2004.07.001
11. K Kashino, K Nakadai, T Kinoshita, H Tanaka, Application of bayesian probability network to music scence analysis, in *Proceedings of IJCAI workshop on CASA*, Montreal, Canada,, 1995, pp. 52–59
12. A Klapuri, Sound onset detection by applying psychoacoustic knowledge, in *Proceedings of ICASSP*, vol. 6. (Phoenix, Arizona, USA, 1999), pp. 3089–3092
13. A Klapuri, Automatic music transcription as we know it today. J New Music Res. **33**(3), 269–282 (2004). doi:10.1080/0929821042000317840
14. JA Moorer, On the segmentation and analysis of continuous musical sound by digital computer, (PhD thesis, Department of Music, Standford University, USA, 1975)
15. B Niedermayer, Non-negative matrix division for the automatic transcription of polyphonic music, in *Proceedings of International Conference on Music Information Retrieval (ISMIR 2008), Austria*, pp. 545–549 (2008)
16. P Smaragdis, JC Brown, Non-negative matric factorization for polyphonic music transcription, in *Proceedings of IEEE workshop Applications of Signal Processing to Audio and Acoustics*, New Paltz, NY, USA, 2003, pp. 177–180
17. S Phon-Amnuaisuk, Transcribing Bach chorales using non-negative matrix factorization, in *Proceedings of the 2010 International Conference on Information Technology Convergence on Audio, Language and Image Processing (ICALIP2010)*, Shanghai China, 2010, pp. 688–693
18. S Sophea, S Phon-Amnuaisuk, Determining a suitable desired factor for non-negative matrix factorisation for polyphonic music transcription, in *Proceedings of the 2007 International Symposium on Information Technology Convergence, (ISITC 2007)*, Sori Arts center, Jeonju, Republic of Korea, 2007, pp. 166–170
19. PJ Walmsley, SJ Godsill, PJW Rayner, Bayesian graphical models for polyphonic pitch tracking, in *Proceedings of diderot forum on mathematics and music*, Vienna, Austria, 1999, pp. 1–26
20. A de Cheveigné, in *Multiple F0 estimation*, ed. by DeLiang W, Brown GJ (Computational Audio Scene Analysis IEEE Press, New York, 2006), pp. 45–79
21. KD Martin, A blackboard system for automatic transcription of simple polyphonic music. M.I.T. Media Lab, Perceptual Computing, Technical Report 385 (1996)
22. I Barbancho, AM Barbancho, A Jurado, LJ Tardón, An information-proach to blind separation and blind deconvolution. Appl Acoust. **65**, 1261–1287 (2004). doi:10.1016/j.apacoust.2004.05.007
23. C Raphael, Aligning music audio with symbolic acores using a hybrid graphical model. Mach Learn. **65**(2-3), 389–409 (2006). doi:10.1007/s10994-006-8415-3
24. M Marolt, A connectionist approach to automatic transcription of polyphonic piano music. IEEE Trans Multimedia. **6**(3), 439–449 (2004). doi:10.1109/TMM.2004.827507
25. E Vincent, X Rodet, Music transcription with ISA and HMM, in *Proceedings of the Fifth International Conference on Independent Component Analysis and Blind Signal Separation (ICA2004)*, Gradana, Spain, 2004, pp. 1197–1204
26. GE Poliner, DPW Ellis, A discriminative model for polyphonic piano transcription. EURASIP J Adv Signal Process. **2007**(1), 154–154 (2007)
27. PO Hoyer, Non-negative sparse coding, in *Proceedings of IEEE Workshop on Neural Networks for Signal Provcessing XII*, Martigny, Switzerland, 2002, pp. 557–565
28. DD Lee, HS Seung, Learning the parts of objects by non-negative matrix factorozation. Nature. **401**, 788–791 (1999). doi:10.1038/44565
29. P Smaragdis, Polyphonic pitch tracking by example in *Proceedings of IEEE workshop Applications of Signal Processing to Audio and Acoustics*, (New paltz, NY, USA, 2011), pp. 125–128

30. DPW Ellis, in *Model-based Scene Analysis*, ed. by DeLiang y, Brown GJ (Computational Audio Scene Analysis IEEE Press, New York, 2006), pp. 115–146
31. DD Lee, HS Seung, in *Algorithms for Non-Negative Matrix Factorization*, ed. by Leen Todd K, Dietterich Thomas G, Volker T (Advances in Neural Information Processing Systems 13 MIT Press, Cambridge, MA, 2001), pp. 556–562
32. J Backus, The Acoustical Foundations of Music, (W.W. Norton & Company, Inc, New York, 1977), 2
33. A Cichocki, R Zdunek, NMFLAB MATLAB Toolbox for non-negative matrix factorization. http://www.bsp.brain.riken.jp/ICALAB/nmflab.html (2006)
34. CL Byrne, Accelerating the EMML algorithm and related iterative algorithms by rescaled block-iterative (RBI) methods. IEEE Trans Image Process. **7**(1), 100–109 (1998). doi:10.1109/83.650854
35. MD Plumbley, SA Abdullah, T Blumensath, ME Davies, Sparse representation of polyphonic music. Signal Process. **86**(3), 417–431 (2005)

doi:10.1186/1687-4722-2012-11
Cite this article as: Phon-Amnuaisuk: **Transcribing Bach chorales: Limitations and potentials of non-negative matrix factorisation.** *EURASIP Journal on Audio, Speech, and Music Processing* 2012 **2012**:11.