LEARNING OPTIMAL FEATURES FOR MUSIC TRANSCRIPTION

Huaiping Ming[1]*, Dongyan Huang*[2]*, Lei Xie*[1] *and Haizhou Li*[2]

[1]School of Computer Science, Northwestern Polytechnical University, Xi'an, China
[2]Institute for Infocomm Research, A*STAR, Singapore

ABSTRACT

This paper aims to design time-frequency representation (TFR) functions for automatic music transcription. It is desirable that the decomposition of those TFR functions are suitable for notes having variation of both pitch and spectral envelop over time. The Harmonic Adaptive Latent Component Analysis (HALCA) model adopted in this paper allows considering those two kinds of variations simultaneously. We evaluate the influence of three TFR functions including IIR, FIR filter bank semigram (FBSG) and constant-Q transform semigram in automatic music transcription task, on a database of popular and polyphonic classic music. The experiment results show that the filter bank based representations are suitable for multiple-instrument recordings and a CQT-based representation turns out to provide very accurate transcription for solo-instrument recordings.

Index Terms— Semigram features, filter bank, constant-Q transform, logarithmic compression, music transcription

1. INTRODUCTION

Automatic music transcription (AMT) aims to convert an audio recording into symbolic representation using musical notation or MIDI file format. Generally, the symbolic representation comprises notes, which are defined by three attributes: the pitches, onsets times and duration of onsets. Therefore, a transcription system consists of notes estimation, multi-pitch detection, and note onset/offset detection. The challenging problem in automatic music transcription is the estimation of concurrent pitches in a time frame, i.e., multiple-F0 or multi-pitch detection. In the past years, the problem of AMT has attracted considerable interest due to numerous applications associated with the area, such as music information retrieval, music archival, computational musicology, and the creation of interactive systems.

Most of AMT systems restrict their scope to performing multi-pitch detection and note tracking. Multi-pitch detection systems were classified according to their estimation type as either joint or iterative [1]. The iterative estimation approach extracts the most prominent pitch in each iteration, however, estimated models tend to accumulate errors at each iteration step, but are computationally inexpensive. On the contrary, joint estimation methods evaluate F0 combinations, leading to more accurate estimates but with increased computational cost.

Audio features of the input time-frequency representation have been used in most multiple-F0 estimation and note tracking either in a joint or an iterative fashion. Typically, a pitch salience function or a pitch candidate set score function is applied to multiple-F0 estimation [2, 3, 1, 4]. More recently, an AMT method based on a mid-level representation was proposed, which derived from a multiresolution Fourier transform combined with an instantaneous frequency estimation and onset detection for computing frame based estimates. A classification-based approach was proposed for piano transcription using features learned from deep belief networks [5] for computing a mid-level time-pitch representation[6].

The multiple-F0 estimation problem is formulated within a statistical framework in many approaches in the literature [7]. The multiple fundamental frequencies and onsets (offsets) can be estimated using the expectation-maximisation (EM) algorithm as MAP estimation in frequency domain [8, 9]. If no prior information is specified, the problem can be expressed as a maximum likelihood (ML) estimation problem using Bayes' rule (e.g. [7, 10]). Some time-domain approaches for multi-pitch detection could be found in the literature [11, 12, 13, 14].

The majority of recent multi-pitch detection papers utilises and expands spectrogram factorization techniques. Non-negative matrix factorization (NMF) is a technique first introduced as a tool for music transcription in [15]. Applications of NMF for AMT include the work by researchers and engineers [16, 17, 18].

An alternative formulation of NMF called probabilistic latent component analysis (PLCA) has also been employed for transcription [19] [20]. Fuentes *et al.* extended the convolutive PLCA algorithm, by modelling each note as a weighted sum of narrowband log-spectra which are also shifted across log-frequency [21]. Three priors of monomodality, sparseness, and spectral envelope temporal continuity are further proposed to enforce the temporal continuity of the timbre of the sources.

Despite significant progress in AMT research, there exists no end-user application that can accurately and reliably transcribe music covering the range of instrument combinations and genres found in recorded music. The performance of even the most recent systems is still clearly far below that of a human expert. It is also worth mentioning that all the new proposed algorithms have not performed better than the algorithm proposed by Yeh and Röbel [1] for multi-pitch estimation since 2009. An algorithm is proposed by Dressler [4] which performs exceptionally well for the task for the two note tracking tasks, bringing the system's performance up to the levels attained for multiple-F0 estimation, but not higher.

Recently, spectrogram factorization becomes more and more popular and could potentially establish itself as the mainstream, although a large number of approaches involve the use of signal processing and feature extraction based techniques. Spectrogram factorization techniques are mainly frame-based even though they can take into account temporal evolution of notes and global signal statistics. The foundation of such technologies is the spectral analysis stage. Likewise, there is no standard method for pre-processing of the signal, with various approaches including the short-time Fourier transform, constant-Q transform [22], filter banks, and auditory models, each leading to different mid-level representations. The challenge in this case is to characterize the impact of such design decisions on AMT results.

This paper focuses on characterizing the impact of filter bank

and CQT semigram feature extraction methods on AMT performance. Audio semigram features, which closely correlate to the aspect of harmony, are a well-established tool in processing and analyzing music data. There are many ways of computing and enhancing semigram features [23, 24], which result in different semigram variants with different properties. But there is no single semigram variant that works well in all applications.

In this paper, we propose to design FIR and IIR filter banks for semigram feature extraction inspiring from IIR filter bank design for chroma feature extraction proposed by Müeller [25]. The proposed filter banks extend the analysed frequency range and features high resolution in time-frequency domain, so the multiple fundamental frequencies in each frame can be well localized. They are suitable for estimating multiple fundamental frequencies of multiple-instrument recordings and simulation results support our hypothesis.

The rest of the paper is organized as follows. The global structure of a music transcription system is presented in Section 2. Then, the semigram representations are described in detail, and the logarithmic compression applied to the features is explained in Section 3. The proposed features are tested and the impact of the logarithmic compression is evaluated in Section 4. Finally we give conclusion and some perspectives in Section 5.

2. STRUCTURE OF A MUSIC TRANSCRIPTION SYSTEM

In an automatic music transcription system that estimates musical note from audio recordings, the input audio signals is often decomposed into small segments on which the pitch of all active notes is estimated. There are many methods proposed for this task [26]. Recently, new methods for smart decomposition of time-frequency representations (TFR) of audio are proposed to address this problem [27, 28, 29, 30, 31]. In the TFR factorization, an audio signal is considered as a sum of basic elements, atoms, or kernels. We use the structure proposed in [32] to evaluate the features, which is suitable for notes having variations of both pitch and spectral envelope overtime.

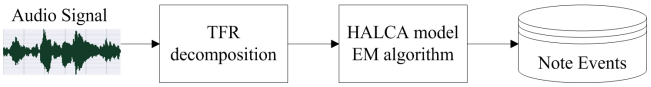

Fig. 1. the structure of music transcription used to evaluate features.

For a given input audio signal, the TFR is first calculated. The TFR decomposition is independently performed from one frame to another, and no assumption of redundancy is made. Then, the TFR is analyzed by the HALCA algorithm, where the input signal is considered as a mixture of several harmonic sources and a noise component. After that, an expectation-maximization (EM) algorithm is applied to estimate all the parameters in the frame work of Probabilistic Latent Component Analysis. Finally, onset/offset detection is employed to transcribe note events. Fig.1 summarizes the global structure of the transcription system we used. In our work, we just switch the TFR decomposition with filter bank Semigram, CQT Semigram and their logarithmic compressed fashion to find optimal features.

3. FILTER BANK SEMIGRAM REPRESENTATIONS

In this section, we introduce the filter bank semigram feature representation functions. The semigram representation in our work is a spectrum representation with logarithmically spaced frequency bins corresponding to one third of the semitone (36 bins per octave) of the musical scale. The Logarithmic Compression which is applied to the feature representation is described as well. In our work, all audio recordings were sampled at 44100Hz and they were resampled to 22050Hz to calculate the feature representations. All the pith representations are calculated considering frequency from A0 (27.5Hz) to A8 (7040Hz) with a time step of 10ms.

3.1. Filter bank semigram

Generally, an array of bandpass filters that separates the input signal into several components is referred to as filter bank [25]. In this work, filter bank is used to decompose an audio signal into frequency bands that correspond to the musical notes. A set of band pass filter was designed, which passes frequency components around the center frequency and rejects all other frequency components for each frequency bin. Since filter design is the basis for further procedure, we need filters that have narrow passbands, sharp cutoffs and high rejection in stopband. To alleviate the filter design requirements, we decide to design filters with different sampling rates. We use a sampling rate of 22050Hz for high pitches (p=96,...,116), 4410Hz for medium pitches (p=60,...,95), and 882Hz for low pitches (p=21,...,59). The reason why we can use lower sampling rates for low pitches is that the time resolution naturally decreases for lower frequencies.

Both IIR filter bank and FIR filter bank are applied in our work. The center frequency of the note A4, 440Hz is used as the reference frequency for all bins. Considering the center frequencies in logarithmic fashion, let *f(b)* denote the center frequency of the bins, then one has the relation

$$f(b) = 2^{\frac{b-207}{36}} * 440, b = 61, ..., 348. \quad (1)$$

The bandwidth of a filter is specified by a *Q factor* of *Q*=25, and the transition band has half the width of the passband.

We use elliptic filters for IIR filter bank design because of their excellent cutoff properties. Each filter is implemented using an eight-order elliptic filter with 1 dB passband ripple and 50dB stopband rejection. For each frequency bin, the bandwidth is given by

$$w = \frac{f(b)}{Q}. \quad (2)$$

From this, we can obtain the cutoff frequencies ω_{p1} and ω_{p2} for the left and right side as

$$\omega_{p1} = f(b) - \frac{w}{2}, \omega_{p2} = f(b) + \frac{w}{2}. \quad (3)$$

So far, all the parameters needed for the IIR filter design are given in absolute terms.

As for FIR filter bank design, a Kaiser window function method is used. The band width of each FIR filter is given by Eq. (2), and the cut off frequency is also given by Eq. (3). To design a FIR filter that has sharp cutoffs, we need to specify the stop and pass frequency at both side of the band pass filter. The stop frequency F_{stop1} and pass frequency F_{pass1} of the left side are given by

$$F_{stop1} = \omega_{p1} - w_{tb}, F_{pass1} = \omega_{p1} + w_{tb}. \quad (4)$$

where w_{tb} is set empirically by: 0.1 for low pitches, 1 for medium pitches and 10 for high pitches. Similarly, the stop frequency F_{stop2} and pass frequency F_{pass2} of the right side are given by

$$F_{stop2} = \omega_{p2} + w_{tb}, F_{pass2} = \omega_{p2} - w_{tb}. \quad (5)$$

Each FIR filter is implemented with 1dB passband ripple, 20dB attenuation in the first stopband and 24dB attenuation in the second stopband. The strictly specified stop and pass frequency will lead to filters with high order (vary from 1295 to 5178 in our work), but the benefit is that we can get a 50dB stopband rejection with relatively low stopband attenuation.

Fig.2 depicts magnitude responses of an N-channel IIR and FIR filter bank, with N = 8, respectively.

After the filter bank is designed, a given audio signal can be decomposed into 288 frequency bins. We compute the short time mean square power using a window of a fixed length and overlap of 50% for each frequency bin. To avoid large phase distortions, a forward-backward filtering is applied so that the resulting output signal has precisely zero phase distortion and a magnitude modified by the square of the magnitude response of filter's [33]. The resulting features, which we denote as filter bank semigram, measure the short time energy content of the audio signal within each frequency bin.

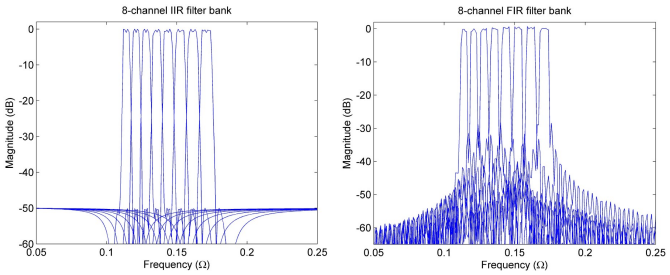

Fig. 2. (a) Magnitude response of an 8-channel IIR filter bank. (b) Magnitude response of an 8-channel FIR filter bank.

3.2. Logarithmic Compression

We applied a logarithmic amplitude compression on the filter bank semigram features in order to account for the human's logarithmic sensation of sound intensity. The energy value e of each frequency bin of semigram feature is replaced by the value $\log(\eta \cdot e + 1)$, where η is a positive constant referred as the compression factor. In our work, the compression factor η is set as $\eta = (5, 10, ..., 100)$ to see whether a logarithmic compression on the features will provides better results.

4. EXPERIMENTS

4.1. Experimental Setup

The semigram representations are tested in an application of music transcription where a Harmonic Adaptive Latent Component Analysis model[32] is used. The onset-time indexes for each note in an audio file are inferred in this application. The Mean Square Error (MSE) between a manually labeled onset-time index (the ground truth) and the detected onset-time index is used for the evaluation of pitch representations. The CQT semigram (we used the implementation provided in [22]) was considered as the contrast features in our experiment. A logarithmic compression on the features as described in section 3.2 is applied, because we need to verify the effect of logarithmic compression and to seek out a proper compression factor η if it provides better results. Generally we got more onsets than the manually labeled onsets, and at few cases we get equal or less onsets than the ground truth. As we need an equal number of onset-time indexes for the computation of MSE, the indexes that nearest the manually labeled indexes were picked out when more onsets were detected, and some onsets may be used more than one times when less onsets were detected.

We select 20 audio recordings from three data sets: Sound Onset Labelizer (SOL) [34], onset detection database (ODB), and NYU data set shared by Professor Juan Pablo Bello at New York University (NYU). This testing corpus of 20 audio recordings includes 10 recordings played by different kind of solo instrument, and 10 recordings in different types of music played with multiple instruments (some with vocals). All the recordings were mono, 16-bit quantified, 44100Hz sampled audio files. Each audio recording has its corresponding manually labeled onset-time indexes as the ground truth. Pitch representations described in section 3.1 and 3.2 were tested on the corpus, and the effect of logarithmic compression described in section 3.3 was explored too.

4.2. Experimental Result

4.2.1. Features comparisons

Fig.3(a) illustrates the sum of MSE of the 10 recordings played by solo instrument against logarithmic compression factors from 0 to 100 (0 means no compression) with an interval of 5. The detailed average MSE of these recordings are shown in the left side of Tab.1. We can clearly see that the CQT semigram [22] get a better result. The FIR and IIR filter bank semigram get approximately the same but relative poor results. Making an insight into each recording, we found the following phenomenon: for the CQT semigram, only two recordings get a little poor result when a relative large logarithmic compression factor is applied, and the other eight recordings always get better results for both compressed and none compressed features. In our test, the MSE result of CQT semigram is within 0.8 and the MSE result of filter bank semigram is within 3.8 for all recording played by solo instrument. This can be explained as follows. Firstly, for both CQT and filter bank semigram features, the number of onsets detected is greater than the number of onsets in the ground truth. That usually means that all the onsets are detected in the recording, so the MSE is relatively small. Secondly, the onsets detected by CQT semigram are much closer to the ground truth than that of filter bank semigram, so the CQT semigram got a better result. We found

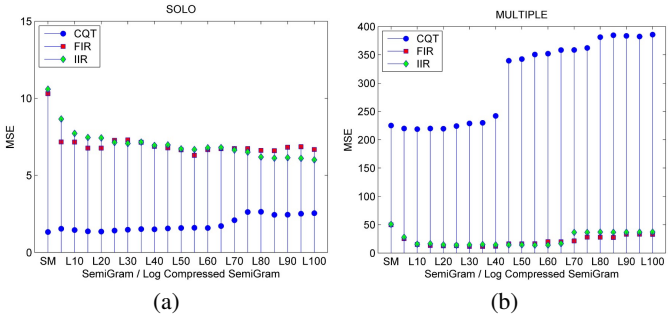

Fig. 3. (a) the sum of MSE of 10 recordings played by solo instrument against logarithmic compression factors (0,5,10,...,100), 0 means none compressed Semigram feature. (b) the sum of MSE of 10 recordings played by multiple instrument against logarithmic compression factors (0,5,10,...,100), 0 means none compressed semigram feature.

that the MSE is less than 10 when the number of onsets detected is greater than the manually labeled for all tested recordings in our experiment. If the number of onsets detected is less than the manually labeled, the MSE is bigger than 10 in most cases. In order to facilitate the analysis, we set the MSE as 100 when it is greater than 100 or there is no onsets detected during the analysis.

Fig.3(b) illustrates the sum of MSE for the 10 recordings played by multiple instruments against logarithmic compression factor from 0 to 100 (0 means no compression) with an interval of 5. The detailed average MSE of these recordings are shown in the right side of Tab.1. As seen in the figure, the FIR and IIR filter bank semigram get a much better result than the CQT semigram. What's more, the FIR and IIR filter bank semigram also get approximately the same results. We found that, only two recordings get a little poor result when there is no logarithmic compression applied or the logarithmic compression factor is small, and the other eight recordings always get better results for compressed and none compressed features. An interesting phenomenon is that, there is no onsets detected for one recording (techno music) by the CQT semigram. In our test, the MSE result of filter bank semigram is within 45 for each recording, and the MSE result of CQT semigram may greater than 1000 or there is no onsets detected for few recordings. We can see that the result getting by filter bank semigram is much stable than that of CQT semigram for multi-instrument recordings. This can be explained as follows. Firstly, for CQT semigram features, the number of onsets detected is smaller than the number of onsets in the ground truth when a large logarithmic compression factor is applied, or there is no onsets detected for some recordings. It is one of the reasons for which the MSE result of CQT semigram varies widely for multi-instrument recordings. Secondly, multiple- instrument (some with vocals) brings much complexity for the onset detection, it results in increase of the MSE result of both FilterBank and CQT semigram.

In terms of feature, the CQT semigram feature is suitable for solo-instrument recordings, and the filter bank semigram feature performs better for multiple-instrument recordings.

4.2.2. Logarithmic Compression Factor

For the logarithmic compression on the CQT semigram, we observe that the compression barely influences the performance for solo-instrument recordings (shown in Fig.1), and a little worse result is obtained when compression factor is larger than 70. The results of multiple-instrument recordings show (in Fig.2) that the logarithmic compression does not bring us good results, even worse results when compression factor is larger than 45.

For the logarithmic compression on the filter bank semigram features, the compression on the features leads to better result for both solo and multiple instruments recordings. Since filter bank semigram is suitable for multiple-instrument recordings, we only make an insight into multiple-instrument recordings. The logarithmic compression gives better result for most recordings, but an interesting phenomenon is that the compression on filter bank semigram feature of an electronic synthesized music (techno music) leads to poor result. This may be caused by the special feature of electronic synthesized music.

All the detailed average MSE of the semigram features is shown in Tab.1. It shows that the onset detection can not be better using logarithmic compression on CQT semigram features, but the performance can be improved using logarithmic compression on filter bank semigram features.

Table 1. the average MSE of three types of features for solo-instrument and multi-instrument recordings.

	Solo Instrument			Multiple Instrument		
	CQT	FIR	IIR	CQT	FIR	IIR
SM	0.132	1.030	1.059	22.51	4.98	5.05
L5	0.154	0.716	0.865	21.99	2.52	2.77
L10	0.146	0.716	0.772	21.88	1.50	1.57
L15	0.137	0.677	0.745	22.00	1.32	1.64
L20	0.136	0.677	0.742	21.96	1.27	1.44
L25	0.142	0.727	0.713	22.41	1.27	1.36
L30	0.147	0.730	0.707	22.89	1.17	1.43
L35	0.151	0.715	0.715	23.00	1.15	1.49
L40	0.150	0.688	0.694	24.22	1.18	1.42
L45	0.156	0.678	0.697	33.96	1.61	1.42
L50	0.159	0.666	0.671	34.26	1.63	1.43
L55	0.160	0.630	0.667	35.06	1.65	1.38
L60	0.159	0.667	0.678	35.21	2.01	1.39
L65	0.171	0.673	0.679	35.85	1.98	1.66
L70	0.210	0.674	0.664	35.85	2.13	3.63
L75	0.263	0.673	0.651	36.21	2.79	3.64
L80	0.264	0.662	0.619	38.12	2.77	3.69
L85	0.245	0.660	0.611	38.45	2.73	3.66
L90	0.245	0.683	0.615	38.35	3.31	3.66
L95	0.251	0.686	0.610	38.23	3.31	3.67
L100	0.255	0.668	0.600	38.57	3.28	3.69

5. CONCLUSIONS

In this paper, we have compared pitch representations including IIR filter bank semigram, FIR filter bank semigram and CQT semigram for music transcription. The effect of a logarithmic compression on the pith representations has been explored as well. Experiments on onset detection have shown that CQT semigram is more suitable for audio recordings played by solo instrument, and filter bank semigram is more suitable for audio recordings played by multiple instruments. A logarithmic compression on CQT semigram does not improve the performance for solo-instrument recordings. But in most cases, a logarithmic compression on filter bank semigram features will lead to better results for multiple instrument recordings. Since the FIR filter bank and IIR filter bank semigram features lead to approximately the same results and the IIR filter bank needs much smaller computation than the FIR filter bank. We conclude that IIR filter bank semigram feature is much suitable for multiple-instrument recordings. In the future, we shall develop more efficient and effective spectrogram factorization techniques for different instrument combinations and genres found in recorded music.

6. ACKNOWLEDGMENTS

This work was supported by a grant from the National Natural Science Foundation of China (61175018) and a grant from the Fok Ying Tung Education Foundation (131059).

7. REFERENCES

[1] C. Yeh, "Multiple fundamental frequency estimation of polyphonic recordings," in *Ph.D. thesis, University Paris VI-Pierre et Marie Curie, France*, 2008.

[2] A. Klapuri, "Multiple fundamental frequency estimation based on harmonicity and spectral smoothness," *IEEE Trans. Au-*

dio, Speech, and Language Processing, vol. 11, pp. 804–816, November 2003.

[3] A. Pertusa and J.M. Inesta, "Multiple fundamental frequency estimation using gaussian smoothness," *Int. Conf. Audio, Speech, and Signal Processing*, pp. 105–108, March 2008.

[4] K. Dressler, "Multiple fundamental frequency extraction for mirex 2012," in *Music Information Retrieval Evaluation eXchange(2012). URL www.music-ir.org/mirex/abstracts/2012/KD1.pdf*, 2012.

[5] E.J. Humphrey, J.P. Bello, and Y. LeCun, "Feature learning and deep architectures: New directions for music informatics," *Journal lf Intelligent Information Systems*, vol. accepted, 2013.

[6] J. Nam, J. Ngiam, H. Lee, and M. Slaney, "A classification-based polyphonic piano transcription approach using learned feature representations," *12th Int. Society for Music Information Retrieval Conf.*, pp. 175–180, 2011.

[7] V. Emiya, R. Badeau, and B. David, "Multipitch estimation of piano sounds using a new probabilistic spectral smoothness principle," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, pp. 1643–1654, August 2010.

[8] H. Kameoka, T. Nishimoto, and S. Sagayama, "A multipitch analyzer based on harmonic temporal structured clustering," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, pp. 982–994, March 2007.

[9] M. Goto, "A real-time music-scene-description system: predominant-f0 estimation for detecting melody and bass lines in real-world audio signals," *Speech Communication 43*, vol. 43, pp. 311–329, 2004.

[10] HA.T. Cemgil, J. Kappen, and D. Barber, "A generative model for music transcription," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 14, pp. 679–694, March 2006.

[11] Z. Duan, B. Pardo, and C. Zhang, "Multiple fundamental frequency estimation by modeling spectral peaks and non-peak regions," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, pp. 2121–2133, February 2010.

[12] A. Koretz and J. Tabrikian, "Maximum a posteriori probability multiple pitch trackiing using the harmonic model," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 19, pp. 2210–2221, 2011.

[13] P. Peeling and S. Godsill, "Multiple pitch estimation using non-homogeneous poisson processes," *IEEE J. Selected Topics in Signal Processing*, vol. 5, pp. 1133–1143, October 2011.

[14] K. Yoshii and M. Goto, "A nonparametric bayesian multipitch analyzer based on infinite latent harmonic allocation," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 20, pp. 717–730, March 2012.

[15] P. Smaragdis and J.C. Brown, "Non-negtive matrix factorization for polyphonic music transcription," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, pp. 177–180, October 2003.

[16] C.T. Lee, Y.H. Yang, and H. Chen, "Multipitch estimation of piano music by exemplar-based sparse representation," *IEEE Trans. Multimedia*, vol. 14, pp. 608–618, June 2012.

[17] N. Bertin, R. Badeau, and E. Vincent, "Enforcing harmonicity and smoothness in bayesian non-negative matrix factorization applied to polyphonic music transcription," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 18, pp. 538–549, March 2010.

[18] K. Ochiai, H. Kameoka, and S. Sagayama, "Explicit beast structure modeling for non-negative matrix factorization-based multipitch analysis," *Int. Conf. Audio, Speech, and Signal Processing*, pp. 133–136, 2012.

[19] P. Smaragdis, B. Raj, and M. Shashanka, "A probabilistic latent variable model for acoustic modeling," *Neural Information Processing Systems Workshop. Whistler, Canada*, 2006.

[20] E. Benetos and S. Dixon, "A shift-invariant latent variable model for automatic music transcripion," *Computer Music Journal*, vol. 36, pp. 81–94, January 2012.

[21] B. Fuentes, R. Badeau, and G. Richard, "Adaptive harmonic time-frequency decomposition of audio using shift-invariant plca," *Int. Conf. Audio, Speech, and Signal Processing*, pp. 401–404, May 2011.

[22] J. Prado, "Une inversion simple de la transformée à q constant," in *http://www.tsi.telecom-paristech.fr/aao/en/2011/06/06/inversible-cqt/*, 2011.

[23] O. Izmirli and R.B. Dannenberg, "Understanding features and distance functions for music sequence alignment," *ISMIR*, pp. 411–416, 2010.

[24] S. Sagayama, K. Takahashi, H. Kameoka, and T. Nishimoto, "Specmurt anasylis: A piano-roll-visualization of polyphonic music signal by deconvolution of log-frequency spectrum," *ISCA Tutorial and Research Workshop (ITRW) on Statistical and Perceptual Audio Processing*, 2004.

[25] M. Müller, *Information retrieval for music and motion*, Springer, 2007.

[26] M. Christensen and A. Jakobsson, *Multi-Pitch Estimation*, Morgan and Claypool Publishers, 2009.

[27] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics*, p. 177180, October 2003.

[28] T. Virtanen, "Monaural sound source separation by nonnegativematrix factorization with temporal continuity and sparseness criteria," *IEEE Trans. on Audio, Speech and Language Processing*, vol. 15, pp. 10661074, March 2007.

[29] D. Lee and H. Seung, "Learning the parts of objects by non-negativity matrix factorization," *Nature*, vol. 401, pp. 788791, October 1999.

[30] C. Févotte, N. Bertin, and J.L. Durrieu, "Nonnegative matrix factorization with the itakura-saito divergence: With application to music analysis," *Neural Computation*, vol. 21, pp. 793830, March 2009.

[31] T. Virtanen, A. T. Cemgil, and S. Godsill, "Bayesian extensions to non-negative matrix factorisation for audio signal modelling," *Proc. of ICASSP*, p. 18251828, March 2008.

[32] B. Fuentes, R. Badeau, and G. Richard, "Harmonic adaptive latent component analysis of audio and application to music transcription," *IEEE Trans. ASLP*, vol. 21(9), pp. 1854–1866, Sept. 2013.

[33] J. G. Proakis and D.G. Manolakis, *Digital Signal Processsing*, Prentice Hall, 1996.

[34] P. Leveau and D. Laurent, "Methodology and tools for the evaluation of automatic onset detection algorithms in music," *Proc. Int. Symp. on Music Information Retrieval, CD-ROM*, 2004.