Music-Information Retrieval in Environments Containing Acoustic Noise

David Grunberg
Department of Electrical and Computer Engineering
Drexel University
3141 Market Street
Philadelphia, Pennsylvania, USA
dgrunberg@drexel.edu

ABSTRACT

In the field of Music-Information Retrieval (Music-IR), algorithms are used to analyze musical signals and estimate high-level features such as tempi and beat locations. These features can then be used in tasks to enhance the experience of listening to music. Most conventional Music-IR algorithms are trained and evaluated on audio that is taken directly from professional recordings with little acoustic noise. However, humans often listen to music in noisy environments, such as dance clubs, crowded bars, and outdoor concert venues. Music-IR algorithms that could function accurately even in these environments would therefore be able to reliably process more of the audio that humans hear. In this paper, I propose methods to perform Music-IR tasks on music that has been contaminated by acoustic noise. These methods incorporate algorithms such as Probabilistic Latent Component Analysis (PLCA) and Harmonic-Percussive Source Separation (HPSS) in order to identify important elements of the noisy musical signal. As an example, a noise-robust beat tracker utilizing these techniques is described.

Categories and Subject Descriptors

H.5.5 [**Information Interfaces and Presentation**]: Sound and Music Computing

General Terms

Design, Experimentation, Performance

Keywords

Music-Information Retrieval, Music, Acoustic Noise

1. INTRODUCTION

Researchers in the field of Music-Information Retrieval, or *Music-IR*, seek to design automated systems to extract high-level features from musical works. Such features, which include beat locations and tempi, can then be used in systems that assist users as they listen to and enjoy music. If a band wishes to incorporate robots or other computer-controlled elements into their performances, for instance, knowledge of beat locations can help ensure that those elements are synchronized with the human musicians. As another example, automatic tempo extraction can help an exerciser choose music with the same speed as their workout, making for a more enjoyable experience. Due in part to its wide utility, this field is now studied by many researchers. The Music Information Retrieval Evaluation eXchange (MIREX), an annual Music-IR competition which has run continuously since 2005, features a vast array of algorithms from all over the world [6].

While this field is advancing rapidly, many of the best Music-IR algorithms and evaluations still focus on noise-free, professional-quality audio. For example, all of the datasets used in the 2013 MIREX beat tracking competition came from clean digital recordings, even the dataset designed to be difficult to track [12]. However, humans listen to much of their music in noisy environments, ranging from crowded bars and dance clubs to outdoor concert venues, and the noise in these environments can contaminate the acoustic signal of the music. As a result, while many Music-IR systems are well adapted to clean, idealized audio, they are not necessarily able to deal with the noise and other distortions present in much of the music humans enjoy.

This work involves the development of noise-robust Music-IR systems. Since the vast majority of existing Music-IR datasets contain only clean audio tracks, methods for collecting datasets of music with various types of acoustic noise are discussed. Subsequently, an algorithm for decomposing an acoustic signal into its component parts is described. Such a system is useful for isolating important elements of noisy musical signals. This algorithm is then used in a noise-robust system for estimating the tempo and the beat locations in a piece of music. This paper finished by exploring some prospective work that could be used to further refine the proposed techniques.

2. RELATED WORK

One method for dealing with noisy audio is to attempt to remove the noise before processing the remaining signal. Methods such as spectral subtraction, in which the noise spectrum is estimated and subtracted from the overall signal, are suboptimal for this purpose because they can create artifacts [1, 2]. As such, techniques were developed that apply gain factors to each spectral bin based on its esti-

MM'14, November 3–7, 2014, Orlando, FL, USA.

http://dx.doi.org/10.1145/2647868.2654858.

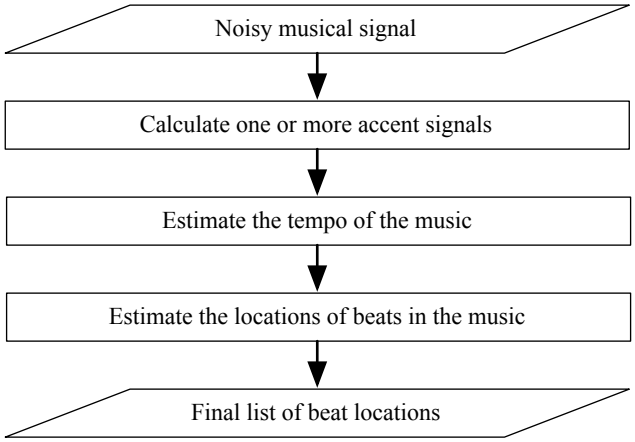

Figure 1: General algorithm for beat tracking.

mated Signal to Noise Ratio (SNR), suppressing only bins that are sufficiently noisy [3]. The result is a cleaner signal with fewer artifacts. However, such noise reduction methods are still imperfect and have difficulty removing all of the noise from a signal. Even with the best systems of this kind, Music-IR algorithms that operate on audio heard in noisy environments will likely have to deal with some noise.

Probabilistic Latent Component Analysis can be used to estimate the content of spectral bins that are obscured by noise [16]. With this method, a set of components that comprise the spectrogram are estimated with an Expectation-Maximization algorithm. Simultaneously, the probability of any particular missing value being formed from a weighted sum of particular components is also estimated. Subjects reported that audio cleaned using this algorithm sounded better than noisy audio not reconstructed with this system.

There are multiple kinds of noise that must be considered when developing a noise-robust algorithm. The noise produced by a moving robot, also called *ego noise*, for instance, is different from that produced by humming lights [13]. Motor noises are nonstationary, as they are based on the position and velocity of the motors, which do not remain constant. This noise may also become somewhat periodic if the robot moves in a repeating pattern, potentially resulting in a situation where the system tracks the robot's noise and not the music. In order to guarantee that a system is noise-robust, particularly in environments incorporating musical robots, Music-IR algorithms designed to be used in noisy environments should be evaluated on audio containing robot noise as well as other noise sources.

In this paper, I give an example algorithm for tempo estimation and beat tracking. This task is generally decomposed into three separate steps (Figure 1) [14]. First, the acoustic waveform of the music is converted into one or more lower-dimensional representations called 'accent signals' which have local extrema at beat locations. Second, the system analyzes the accent signal(s) to estimate the tempo of the music. Finally, the system uses the estimated tempo and accent signal(s) to determine beat locations. While beat tracking systems vary in terms of the individual algorithms and features, most of them utilize these three steps [9].

3. PROPOSED APPROACH

My goal is to design Music-IR algorithms that can perform accurately not just on audio that is taken directly from clean, professionally-produced digital recordings, but also on music recorded from noisy acoustic channels. By obtaining noisy data and building algorithms that can reliably function on that music, I hope to create Music-IR tools that function on more of the music that humans enjoy.

One desired property of my final system is that it be robust to multiple types of noise. This will allow for it to be more flexible than if it were only robust to specific noise sources. I therefore am recording datasets of audio contaminated by a variety of different noise sources. This will allow me to evaluate how my system performs on conventional noise signals such as the humming of lights as well as more difficult noise sources such as the ego noise produced by robots [13]. I am also annotating the music in these datasets with ground-truth values for the high-level features that I am searching for, such as tempo and beat locations, in order to better evaluate the accuracy of my algorithms.

When designing a system that is to process noisy data, there are two general approaches. One is to try to remove as much of the noise as possible from the signal and then process the signal normally, and the other is to try to process the noisy signal in some manner that is robust to noise. The former approach has the disadvantage that noise reduction algorithms are both unlikely to remove all of the noise, and can introduce distortions into the signal [2]. As such, I am building systems that, rather than removing noise, are designed to performed well even if the audio is noisy.

In summary, I am gathering annotated datasets of audio contaminated with multiple types of noise, and then designing noise-robust Music-IR algorithms that will be tested on those datasets. I am using Probabilistic Latent Component Analysis (PLCA) and Harmonic-Percussive Source Separation (HPSS) in order to find important information from the signals even in the presence of noise. I have already designed a beat-tracking algorithm that uses these methods to calculate possible accent signals. Subsequently, my beat tracker uses a periodicity estimator and a dynamic programming algorithms to obtain the tempo and beat estimates.

4. CONTRIBUTIONS

My contributions thus far can be grouped into two areas. First, I collected multiple datasets of musical audio contaminated with various types of noise. Second, I designed a beat tracking algorithm that is robust to the noises in the datasets. Both aspects of my work are important for the continued pursuit of noise-robust Music-IR in general.

4.1 Datasets

My first step was to gather datasets of noisy audio. I collected twenty songs with clearly identifiable beats as my baseline set of 'clean' songs and verified that the songs could be accurately tracked by conventional beat trackers. This helped to ensure that, if my tracker performed poorly on the music after I added noise, the low accuracies would be due to the noise and not the music itself. I then manually marked all of the beats in each song. This provided me with ground-truth data that I could use to evaluate my system.

The next step was to add noise sources to the audio. In order to set up a recording apparatus in a noisy environment, I mounted microphones on and around a Hubo robot (Figure

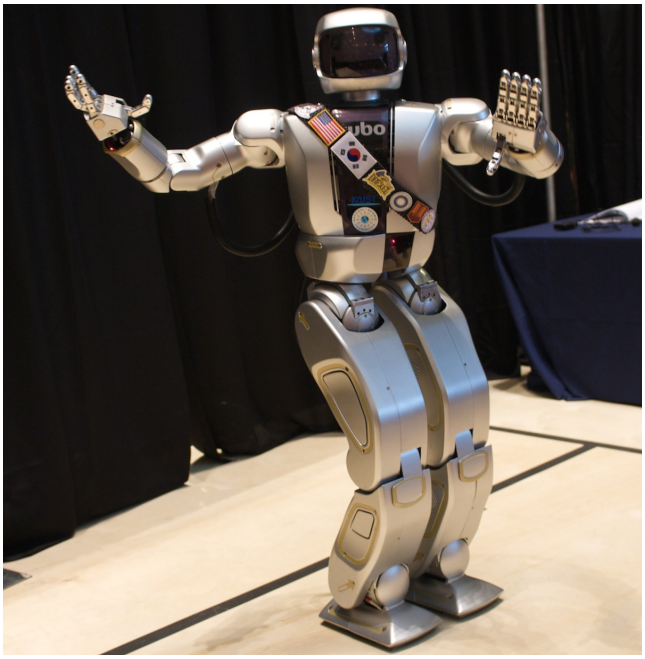

Figure 2: A Hubo robot.

2). Hubo was used in order to provide additional sources of noise that would not be present in an ordinary empty room. I played the clean music through a speaker and recorded it on the microphones, taking one dataset in which the robot was on but unmoving, and others in which the robot was moving its arms either randomly, in a predetermined pattern, or in synchrony with the music. The first dataset contained noise from the robot's fans and computers, in addition to sounds provided by the room itself such as lighting hums. The other datasets added ego noise as well. Finally, the ground-truth annotations were added to both sets of noisy music tracks.

4.2 Algorithm

My beat tracking algorithm begins by taking five seconds of audio and calculating its magnitude spectrum (Figure 3). An implementation of HPSS is then performed on the spectrum [8]. Beats are often more percussive than harmonic in nature, so HPSS allows the system to discard harmonic data and use only the more meaningful percussive component.

The percussive component is then processed by PLCA, which uses an Expectation-Maximization Algorithm to decompose the audio into 20 components [10, 16]. This method estimates the components and their activation probabilities over time. A component with a periodic beat will likely have large activation values spaced at that period and small values elsewhere. As such, the activations of the components are potential accent signals for estimating beat positions. I determined that activations of individual components would be more likely to provide reliable information in noisy environments than conventional beat tracking features such as spectral contrast or spectral flux, which are evaluated over the entire audio signal (e.g., the mixture of all the components) and so may be distorted by noise [9]. I thus chose to use PLCA, which can accurately identify such components, for the decomposition step.

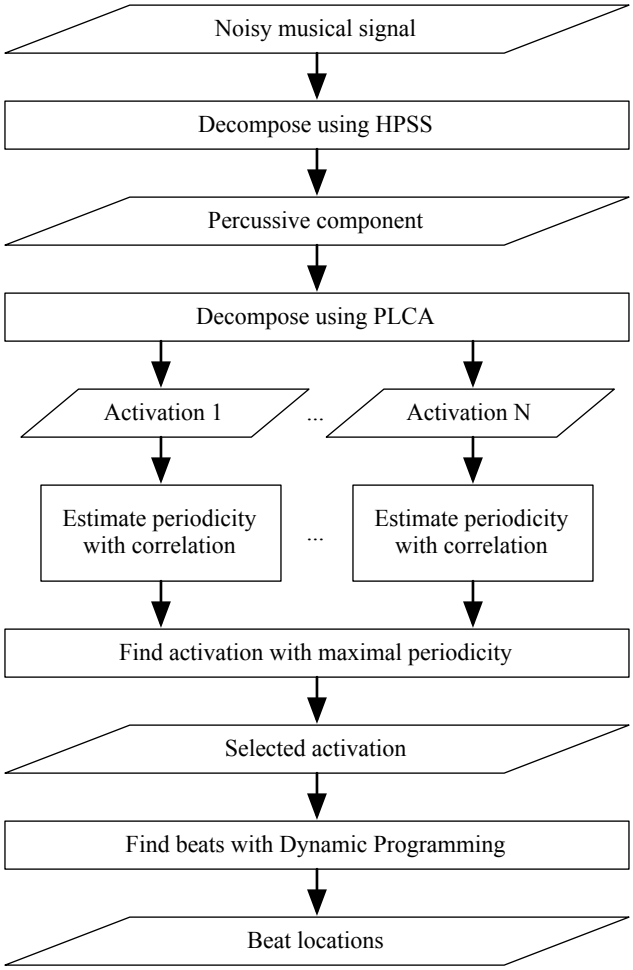

Figure 3: The beat tracking algorithm.

The activations for each component are then correlated with trains of impulses spaced at intervals corresponding to various tempi. The activation signal with the highest correlation is considered to be the most periodic, and thus the best accent signal of the group. The period of the impulse train at that correlation is used to estimate the tempo of the music. From there, dynamic programming is used to find the best sequence of beats spaced according to that tempo and occurring at local maxima of the accent signal.

4.3 Results

In order to determine the negative effects of noise, I evaluated my noisy audio datasets on a beat tracker not designed for noise robustness [11]. As expected, performance dropped sharply. The clean dataset achieved an average F-Measure score of about .98, very close to the maximal value of 1. The dataset taken with a randomly moving robot, however, did no better than .84. Spectral subtraction not only failed to bring accuracy back up to .98, but also required substantial fine-tuning that limited its general utility. This demonstrated how deleterious noise can be on the performance of conventional beat trackers. It also indicated that a noise-removal approach was unlikely to work as well as I wished, so a noise-robust algorithm could be preferable.

Table 1: Average beat tracker accuracy for various metrics on noisy musical audio.

	Dataset 1 (stationary robot)			Dataset 2 (moving robot)		
Tracker	FMeasure	Information Gain	CMLc	FMeasure	Information Gain	CMLc
Proposed	**91.4**	**91.6**	**90.1**	**82.5**	**75.4**	**77.8**
Autocorrelation	82.9	78.8	79.5	77.1	65.5	70.5
Tracker 1 [5]	61.1	55.9	56.5	55.7	48.5	41.2
Tracker 2 [7]	77.1	47.1	40.4	76.3	44.7	41.2
Tracker 3 [15]	60.9	48.1	27.7	58.5	44.5	27.0

I also evaluated my beat tracker algorithm on the noisy audio. My tracker was evaluated on two of my audio datasets: one in which the robot did not move (denoted *Dataset 1*) and one taken while Hubo moved its arms periodically (*Dataset 2*). Its performance is compared using three standard beat tracking metrics ([4]) with a similar system that uses autocorrelation instead of correlation for the periodicity step, as well as with three state-of-the-art trackers, in Table 1 [5, 7, 15]. These three trackers were selected for use in a multiple-tracker system that was restricted to only 5 trackers, confirming their utility [12]. Even on the harder dataset, the proposed system still surpassed the other systems. More results are detailed at [10].

5. PLANNED WORK

I plan to advance this work in three main areas. The first is expanding the dataset by using music from a wider variety of artists and genres and by re-recording the audio in environments containing different types of noise. This will help validate the robustness of my algorithm.

I will also enable my algorithm to utilize multiple components from the acoustic signal. These components could be used to help determine multiple rhythmic sources in a single piece of music, and could also help categorize different types of beats. If a piece used a bass drum to play the onbeats and a hi-hat to play the other beats, for example, then by examining the PLCA components any particular beat could be identified as belonging to the appropriate category.

Finally, I will develop a program for the Hubo that uses the beats estimated by my algorithm in order to respond to live music, such as by tapping along. This will be another step towards enabling robots to respond to music. It will also be a suitable demonstration of the utility of my work.

6. ACKNOWLEDGEMENTS

This work was supported by NSF grant #CNS-0960061 MRI-R2: Development of a Common Platform for Unifying Humanoids Research and by a NSF Graduate Research Fellowship.

7. REFERENCES

[1] S. Boll. Suppresion of acoustic noise in speech using spectral subtraction. *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 27(2):113–120, 1979.

[2] O. Cappe. Elimination of the musical noise phenomenon with the ephraim and malah noise suppressor. *IEEE Transactions on Speech and Audio Processing*, 2(2):345–349, April 1994.

[3] I. Cohen. Speech enhancement using a noncausal a priori SNR estimator. *IEEE Signal Processing Letters*, 11(9):725–728, September 2004.

[4] M. Davies, N. Degara, and M. Plumbley. Evaluation methods for musical audio beat tracking algorithms. Technical Report C4DM-TR-09-06, Queen Mary University of London: Center for Digital Music, 2009.

[5] S. Dixon. Evaluation of the audio beat tracking system beatroot. *Journal of New Music Research*, 36(1):39–50, March 2007.

[6] J. S. Downie. The music information retrieval evaluation exchange (2005-2007): A window into music information retrieval research. *Acoustical Science and Technology*, 29(4):247–255, 2008.

[7] D. P. W. Ellis. Beat tracking by dynamic programming. *Journal of New Music Research*, 36(1):51–60, 2007.

[8] D. FitzGerald. Harmonic/percussive separation using median filtering. In *Proceedings of the International Conference on Digital Audio Effects*, 2010.

[9] F. Gouyon, S. Dixon, and G. Widmer. Evaluating low-level features for beat classification and tracking. In *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 4, pages 1309–1312, Honolulu, USA, April 2007.

[10] D. Grunberg and Y. Kim. Rapidly learning musical beats in the presence of environmental and robot ego noise. In *Proceedings of the International Conference on Intelligent Robots and Systems (in press)*, 2014.

[11] D. Grunberg, D. Lofaro, P. Oh, and Y. Kim. Robot audition and beat identification in noisy environments. In *Proceedings of the International Conference on Intelligent Robots and Systems*, pages 2916–2921, 2011.

[12] A. Holzapfel et al. Selective sampling for beat tracking evaluation. *IEEE Transactions on Audio, Speech, and Language Processing*, 20(9):2539–2548, 2012.

[13] G. Ince et al. Robust ego noise suppression of a robot. In *Trends in Applied Intelligent Systems, LNAI 6096, Lecture Notes in Computer Science*, pages 62–71. Springer-Verlag, June 2010.

[14] M.-Y. Kao et al. Tempo and beat tracking for audio signals with music genre classification. *International Journal of Intelligent Information and Database Systems*, 3(3):275–290, August 2009.

[15] J. L. Oliveira et al. IBT: A real-time tempo and beat tracking system. In *Proc. of the 2010 International Society on Music Information Retrieval*, pages 291–296, 2010.

[16] P. Smaragdis, B. Raj, and M. Shashanka. Missing data imputation for spectral audio signals. In *Proceedings of the IEEE International Workshop on Machine Learning for Signal Processing*, 2009.