Finding Meter in Music Using an Autocorrelation Phase Matrix and Shannon Entropy

Douglas Eck
University of Montreal
Department of Computer Science
CP 6128, Succ. Centre-Ville
Montreal, Quebec H3C 3J7 CANADA
eckdoug@iro.umontreal.ca

Norman Casagrande
University of Montreal
Department of Computer Science
CP 6128, Succ. Centre-Ville
Montreal, Quebec H3C 3J7 CANADA
casagran@iro.umontreal.ca

ABSTRACT

This paper introduces a novel way to detect metrical structure in music. We introduce a way to compute autocorrelation such that the distribution of energy in phase space is preserved in a matrix. The resulting autocorrelation phase matrix is useful for several tasks involving metrical structure. First we can use the matrix to enhance standard autocorrelation by calculating the Shannon entropy at each lag. This approach yields improved results for autocorrelation-based tempo induction. Second, we can efficiently search the matrix for combinations of lags that suggest particular metrical hierarchies. This approach yields a good model for predicting the meter of a piece of music. Finally we can use the phase information in the matrix to align a candidate meter with music, making it possible to perform beat induction with an autocorrelation-based model. We present results for several meter prediction and tempo induction datasets, demonstrating that the approach is competitive with models designed specifically for these tasks. We also present preliminary beat induction results on a small set of artificial patterns.

Keywords: Meter prediction, tempo induction, beat induction, autocorrelation, entropy

1 Introduction

In this paper we introduce an autocorrelation phase matrix, a two-dimensional structure (computed from MIDI or digital audio) that provides the necessary information for estimating the lags and phases of the music's metrical hierarchy. We use this matrix as the core data structure to estimate the meter of a piece (meter prediction), to estimate the tempo of a piece (tempo induction) and to align the piece of music with the predicted metrical structure (beat induction).

We will provide algorithm details and experimental results for meter prediction and tempo induction. We will also present some details concerning the alignment of the metrical structure with a piece of music. We will also present alignment results for a small dataset of artificial patterns. However the details of computing this alignment online (for beat induction) are the topic of another paper.

The structure of this paper is as follows. In Section 2 we will discuss other approaches to finding meter and beat in music. In Section 3 we will describe our model consisting of the creation of an autocorrelation matrix, computation of the entropy for each lag in this matrix, the selection of a metrical hierarchy and the alignment of the hierarchy with music. Finally in Section 4 we present simulation results.

Due to space constraints we have omitted details for aligning the autocorrelation phase matrix with a musical signal so as to aid in beat induction. A longer report containing these details is available at www.iro.umontreal.ca/~eckdoug/publications.html.

2 Meter and Autocorrelation

Meter is the sense of strong and weak beats that arises from the interaction among hierarchical levels of sequences having nested periodic components. Such a hierarchy is implied in Western music notation, where different levels are indicated by kinds of notes (whole notes, half notes, quarter notes, etc.) and where bars establish measures of an equal number of beats (Handel, 1993). For instance, most contemporary pop songs are built on four-beat meters. In such songs, the first and third beats are usually emphasized. Knowing the meter of a piece of music helps in predicting other components of musical structure such as the location of chord changes and repetition boundaries (Cooper and Meyer, 1960).

Autocorrelation works by transforming a signal from the time domain into the frequency domain. Autocorrelation provides a high-resolution picture of the relative salience of different periodicities, thus motivating its use in tempo and meter related music tasks. However, the autocorrelation transform discards all phase information, making it impossible to *align* salient periodicities with the music. Thus autocorrelation can be used to predict, for example, that music has something that repeats every 1000ms but it cannot say *when* the repetition takes place

relative to the start of the music. One primary goal of our work here is to compute autocorrelation efficiently while at the same time preserving the phase information necessary to perform such an alignment. Our solution is the autocorrelation phase matrix.

Autocorrelation is certainly not the only way to perform meter prediction and related tasks like tempo induction. Adaptive oscillator models (Large and Kolen, 1994; Eck, 2002) can be thought of as a time-domain correlate to autocorrelation based methods and have shown promise, especially in cognitive modeling. Multi-agent systems such as those by Dixon (2001) have been applied with success. as have Monte-Carlo sampling (Cemgil and Kappen, 2003) and Kalman filtering methods (Cemgil et al., 2001).

Many researchers have used autocorrelation for music information retrieval. Due to space constraints only a short listing is provided here. Brown (1993) used autocorrelation to find meter in musical scores represented as note onsets weighted by their duration. Vos et al. (1994) proposed a similar autocorrelation method. The primary difference between their work and that of Brown was their use of melodic intervals in computing accents. Scheirer (1998) provided a model of beat tracking that treats audio files directly and performs relatively well over a wide range of musical styles (41 correct of 60 examples). Volk (2004) explored the influence of interactions between levels in the metrical hierarchy on metrical accenting. Toiviainen and Eerola (2004) also investigated an autocorrelation-based meter induction model. Their focus was on the relative usefulness of durational accent and melodic accent in predicting meter. Klapuri et al. (2005) incorporate the signal processing approaches of Goto (2001) and Scheierer in a model that analyzes the period and phase of three levels of the metrical hierarchy.

3 Model Details

3.1 Preprocessing

For MIDI files, the onsets can be transformed into spikes with amplitude proportional to their midi note onset volume. Alternately MIDI files can simply be rendered as audio and written to wave files. Stereo audio files are converted to mono by taking the mean of the two channels. Then files are downsampled to some rate near 1000Hz. The actual rate is kept variable because it depends on the original sampling rate. For CD-audio (44.1Khz), we used a sampling rate of 1050Hz allowing us to downsample by a factor of 42 from the original file. Best results were achieved by computing a sum-of-squares envelope over windows of size 42 with 5 points of overlap. However for most audio sources a simple decimation and rectification works as well. The model was not very sensitive to changes in sampling rate nor to minor adjustments in the envelope computation such as substituting RMS (root mean square) for the sum of squares computation.

3.2 Autocorrelation Phase Matrix

Autocorrelation is a special case of cross-correlation where $x_1 == x_2$. There is a strong and somewhat surprising link between autocorrelation and the Fourier transform. Namely the autocorrelation A of a signal X (having length N) is:

$$A(X) \quad = \quad ifft(|fft(X)|) \tag{1}$$

where fft is the (fast) Fourier transform, $ifft$ is the inverse (fast) Fourier transform and $||$ is the complex modulus. One advantage of autocorrelation for our purposes is that it is defined over periods rather than frequencies (note the application of the IFFT in Equation 1), yielding better representation of low-frequency information than is possible with the FFT.

Autocorrelation values for a random signal should be roughly equal across lags. Spikes in an autocorrelation indicate temporal order in a signal, making it possible to use autocorrelation to find the periods at which high correlation exists in a signal. As a music example, consider the autocorrelation for a ChaChaCha from the ISMIR 2004 Tempo Induction contest is shown (Figure 1). The peaks of the autocorrelation align with the tempo and integer multiples of the tempo.

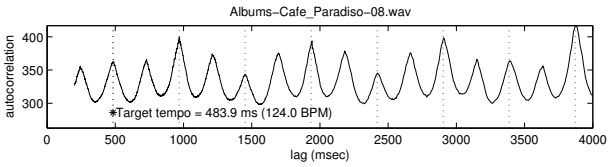

Figure 1: Autocorrelation of a ChaChaCha from the ISMIR 2004 Tempo Induction contest (Albums-Cafe_Paradiso-08.wav). The dotted vertical lines mark the actual tempo of the song (484 msec, 124 bpm) and harmonics of the tempo.

Unfortunately autocorrelation has been shown in practice to not work well for many kinds of music. For example when a signal lacks strong onset energy, as it might for voice or smoothly changing musical instruments like strings, the autocorrelation tends to be flat. See for example a song from Manos Xatzidakis from the ISMIR 2004 Tempo Induction in Figure 2. Here the peaks are less sharp and are not well-aligned with the target tempo. Note that the y-axis scale of this graph is identical to that in Figure 1.

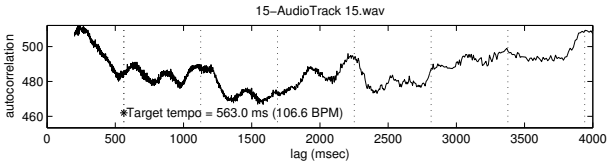

Figure 2: Autocorrelation of a song by Manos Xatzidakis from the ISMIR 2004 Tempo Induction contest (15-AudioTrack 15.wav). The dotted vertical lines mark the actual tempo of the song (563 msec, 106.6 bpm) and harmonics of the tempo.

One way to address this is to apply the autocorrelation to a number of band-pass filtered versions of the signal, as discussed in Section 3.1. In place of multi-band processing we compute the distribution of autocorrelation energy in phase space. This has a sharpening effect, allowing autocorrelation to be applied to a wider range of signals than autocorrelation alone without extensive preprocessing.

The autocorrelation phase information for lag l is a vector A_l:

$$A_l = \left(\sum_{i=0}^{\lfloor \frac{N-l}{l} \rfloor} x_{li+\phi} \; x_{l(i+1)+\phi} \right)_{\phi=0}^{l-1} \quad (2)$$

We compute an autocorrelation phase vector A_l for each lag of interest. In our case the minimum lag of interest was 200ms and the maximum lag of interest was 3999ms. Lags were sampled at 1ms intervals yielding $L = 3800$ lags. Equation 2 effectively "wraps" the signal modulo the lag l question, yielding vectors of differing lengths ($|A_l| == l$). To simplify later computations we normalized the length of all vectors by resampling. This was achieved by fixing the number of phase points for all lags at K ($K = 50$ for all simulations; larger values were tried and yielded similar results but significantly smaller values resulted in a loss of temporal resolution) and resampling the variable length vectors to this fixed length. This process yielded an autocorrelation phase matrix P where $|P| = [L, K]$.

To provide a simple example, we use the first pattern from the set found in Povel and Essens (1985). See Section 4.4 for a description of how these patterns are constructed. For this example we set the base inter-onset interval to be 300ms. In Figure 3 the autocorrelation phase matrix is shown. On the right, the sum of the matrix is shown. It is the standard autocorrelation.

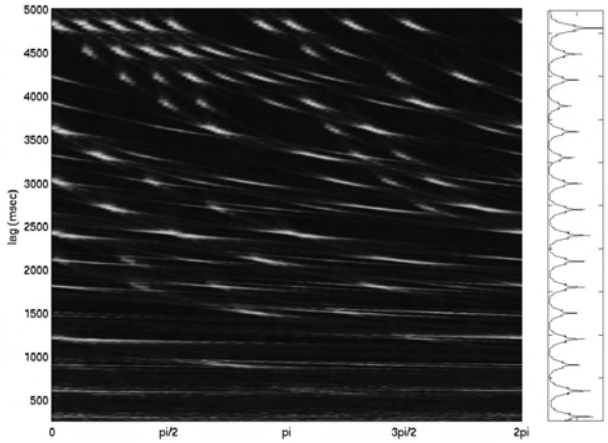

Figure 3: The autocorrelation phase matrix for Povel & Essens Pattern 1 computed for lags 250ms through 500ms. The phase points are shown in terms of relative phase $(0, 2\pi)$. Black indicates low value and white indicates high value. Since only relative values are important, the exact colormap is not shown. On the right, the autocorrelation is displayed; it was recovered by taking the row-wise sum of the matrix.

3.3 Shannon Entropy

As already discussed, is possible to improve significantly on the performance of autocorrelation by taking advantage of the distribution of energy in the autocorrelation phase matrix. The idea is that metrically-salient lags will tend to be have more "spike-like" distribution than non-metrical lags. Thus even if the *autocorrelation* is evenly distributed by lag, the *distribution of autocorrelation energy in phase space* should not be so evenly distributed. There are at least two possible measures of "spikiness" in a signal, variance and entropy. We focus here on entropy, although experiments using variance yielded very similar results.

Entropy is the amount of "disorder" in a system. Shannon entropy H:

$$H(X) = -\sum_{i=1}^{N} X(i) log_2[X(i)] \quad (3)$$

where X is a probability density.

We compute the entropy for lag l in the autocorrelation phase matrix by as follows:

$$A_{sum} = \sum_{i=0}^{N} A_l(i) \quad (4)$$

$$H_l = -\sum_{i=0}^{N} A_l(i)/A_{sum} log_2[A_l(i)/A_{sum}] \quad (5)$$

This entropy value, when multiplied into the autocorrelation, significantly improves tempo induction. For example, in Figure 4 we show the autocorrelation along with the autocorrelation multiplied by the entropy for the same Manos Xatzidakis show in in Figure 2. On the bottom observe how the detrended (1- entropy) information aligns well with the target lag and its multiples. Detrending was done to remove a linear trend that favors short lags. (Simulations revealed that performance is only slightly degraded when detrending is omitte.) Most robust performance was achieved when autocorrelation and entropy were multiplied together. This was done by scaling both the autocorrelation and the entropy to range between 0 and 1 and then multiplying them together.

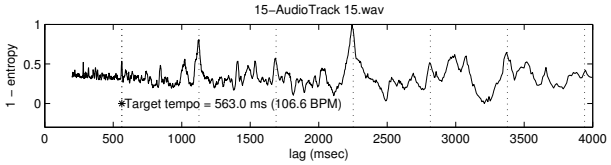

Figure 4: Entropy-of-phase calculation for the same Manos Xatzidakis song shown in Figure 2. The plot displays (1 - entropy), scaled to $[0, 1]$ and detrended. Observe how the entropy spikes align well with the correct tempo lag of 563ms and with its integer multiples (shown as vertical dotted lines). Entropy compares favorably with the raw autocorrelation of the same song as shown in Figure 2.

3.4 Metrical hierarchy selection

We now move away from the autocorrelation phase matrix for the moment and address task of selecting a winning metrical hierarchy. A rough estimate of meter can be had by simply summing hierarchical combinations of autocorrelation lags. In place of standard autocorrelation we use the product of autocorrelation and (1 - entropy) AE as described above. The likelihood of a duple meter M^{duple}

existing at lag l can be estimated using the following sum:

$$M_l^{duple} = AE(l) + AE(2l) + AE(4l) + AE(8l) \quad (6)$$

The likelihood of a triple meter is estimated using the following sum:

$$M_l^{triple} = AE(l) + AE(3l) + AE(6l) + AE(12l) \quad (7)$$

Other candidate meters can be constructed. using similar combinations of lags. A winning meter can be chosen by sampling all reasonable lags (e.g. $200ms <= l <= 2000ms$) and comparing the resulting M_l^* values. Provided that the same number of points are used for all candidate meters, these M_l^* values can be compared directly, allowing for a single winning meter to be selected among all possible lags and all possible meters. Furthermore, this search is efficient given that each lag/candidate meter combination requires only a few additions. For the meter prediction simulations in Section 4 this was the process used to select the meter.

3.5 Prediction of tempo

Once a metrical hierarchy is chosen, there are several simple methods for selecting a winning tempo from among the winning lags. One option is to pick the lag closest to a comfortable tapping rate, say 600ms. A second better option is to multiply the autocorrelation lags by a window such that more accent is placed on lags near a preferred tapping rate. The window can be applied either before or after choosing the hierarchy. If it is applied before selecting the metrical hierarchy, then the selection process is biased towards lags in the tapping range. We tried both approaches; applying the window before selection yields better results, but only marginally better (on the order of 1% better performance on the tempo prediction tasks described below). To avoid adding more parameters to our model we did not construct our own windowing function. Instead we used the function (with no changes to parameters) described in Parncutt (1994): a Gaussian window centered at 600ms and symmetrical in log-scale frequency.

4 Simulations

We have run the model on several datasets. To test tempo induction we used the Ballroom and Song Excerpts databases from the ISMIR 2004 Tempo Induction contest. For testing the ability of the model to perform meter prediction we used the the Essen European Folksong database and the Finnish Folk Song database. We also include preliminary simulations on alignment using the 35 artificial patterns from Povel and Essens (1985) as well as

4.1 ISMIR 2004 Tempo Induction

We used two datasets from the ISMIR 2004 Tempo Induction contest (Gouyon et al., 2005). The first dataset was the Ballroom dataset consisting of 698 wav files each approximately 30 seconds in duration encompassing eight musical styles. See Table 1 for a breakdown of song styles

Table 1: Performance of model by genre on the Ballroom dataset. See text for details.

Style	Count	Acc. A	Acc. B	Acc. C
ChaChaCha	111	106	107	109
Jive	60	6	60	60
Quickstep	82	0	77	80
Rumba	98	84	85	92
Samba	86	78	79	83
Tango	86	81	82	83
Vienn.Waltz	65	0	57	64
Waltz	110	86	86	93
Global	698	441	633	664

Table 2: Summary of models on the Ballroom dataset. See text for details.

Model	Acc. A	Acc. B	Acc. C
Acorr Only	49%	77%	77%
Acorr+Meter	58%	80%	85%
Acorr+Entropy	41%	85%	85%
Full Model	**63%**	**91%**	**95%**
Klapuri	**63%**	**91%**	93%

along with the performance of our model on the dataset. In the table, "Acc. A" is Accuracy A from the contest: the number of correct predictions within 4% of the target tempo. "Acc. B" is Accuracy B from the contest. It also takes into account misses due to predicting the wrong level of the metrical hierarchy. Thus answers are treated as correct if they are within 4% of the target tempo multiplied by 2,3,1/2 or 1/3. "Acc C." is our own measure which also treats answers as correct if they are within 4% of the target tempo multiplied by 2/3 or 3/2. This gives us a measure of model failure due to predicting the wrong meter.

We computed several baseline models for the ballroom dataset. These results are shown along with our best results and those of the contest winner, Klapuri et al. (2005), in Table 2. The "Acorr Only" model uses simple autocorrelation. The "Acorr+Meter" model incorporates the strategy described in this paper for using multiple hierarchically-related lags in prediction. The "Acorr+Entropy" uses autocorrelation plus entropy as computed on the phase autocorrelation matrix (but no meter). The full model could also be called "Acorr+Entropy+Meter" and is the one described in this paper. "Klapuri" shows the results for the contest winner.

Two things are important to note. First, it is clear that both of our two main ideas, meter reinforcement ("Meter") and entropy calculation ("Entropy") aid in computing tempo. Second, the model seems to work well, returning results that compete with the contest winner.

We also used the "Song Excerpts" dataset from the ISMIR 2005 dataset. This dataset consisted of 465 songs of roughly 20sec duration spanning nine genres. Due to space constraints, we do not report model performance on individual genres. In table Table 3 the results are summarized in a format identical to Table 2.

Table 3: Summary of models on the Song Excerpts dataset. See text for details.

Model	Acc. A	Acc. B	Acc. C
Acorr Only	49%	64%	64%
Acorr+Meter	50%	80%	85%
Acorr+Entropy	53%	74%	74%
Full Model	**60%**	79%	88%
Klapuri	58%	**91%**	**94%**

Here it can be seen that our model performed slightly better than the winning model on Accuracy A but performed considerably worse on Accuracy B. In our view, Accuracy B is a more important measure because it reflects that the model has correctly predicted the metrical hierarchy but has simply failed to report the appropriate level in the hierarchy.

4.2 Essen Database

We computed our model on a subset of the Essen collection (Schaffrath, 1995) of European folk melodies. We selected all melodies in either duple (i.e. having 2^n eighth notes per measure; e.g. 2/4 and 4/4) or triple/compound meter (i.e having 3n eighth notes per measure; e.g. 3/4 and 6/8). This resulted in a total of 5507 melodies of which 57% (3121) were in duple meter and 43% (2386) were in triple/compound meter. The task was to predict the meter of the piece as being either duple or triple/compound. This is exactly the same dataset and task studied in Toiviainen and Eerola (2004).

Our results were promising. We classified 90% of the examples correctly (4935 of 5507 correct). Our model performed better on duples than triple/compounds, classifying 94% of the duple examples correctly (2912 of 3121 correct) and 85% of the triple/compound examples correctly (2023 of 2386 correct).

These success rates are similar to those in Toiviainen and Eerola (2004). However it is difficult to compare our approaches because their data analysis technique (stepwise discriminant function analysis) does not control for in-sample versus out-of-sample errors. Functions are combined using the target value (the meter) as a dependent variable. This is suitable for weighing the relative predictive power of each function but not suitable for predicting how well the ensemble of functions would perform on unseen data unless training and testing sets or cross-validation is used. Our approach used no supervised learning.

4.3 Finnish Folk Songs Database

We performed the same meter prediction task on a subset of the Finnish Folksong database (Eerola and Toiviainen, 2004). This dataset was also treated by Toiviainen and Eerola (2004) and the selection criteria were the same. For this dataset we used 7139 melodies of which 80% (5720) were in duple meter and 20% (1419) were triple/compound meter. (For the Toiviainen et. al. study, 6861 melodies were used due to slightly more stringent selection criteria. However the ratio of duples to triple/compounds is almost identical.) Note that the datasets are seriously imbalanced: a classifier which always guesses duple will have a success rate of 80%. However given the relative popularity of duple over triple, this imbalance seems unavoidable.

Our results were promising. We classified 93% examples correctly (6635 of 71239 correct). Again, our model performed better on duples than triple/compounds, classifying 95% of the duple examples correctly (5461 of 5720 correct) and 83% of the triple/compound examples correctly (1174 of 1419 correct).

4.4 Povel & Essens Patterns

To test alignment (beat induction) we used a set of rhythms from Experiment 1 of Povel and Essens (1985). These rhythms are generated by permuting the interval sequence **1 1 1 1 1 2 2 3** and terminating it by the interval **4**. These length-16 patterns all contain nine notes and seven rests.

Their model works by applying a set of rules that forced the accentuation of (a) singleton isolated events, (b) the second of two isolated events and (c) the first and last of a longer group of isolated events. Of particular importance is that they validated their model using a set of psychological experiments with human subjects.

Our model predicted the correct downbeat (correct with respect to the Povel & Essens model) 97% of the time (34 of 35 patterns). The pattern where the model failed was pattern 27. Our interest in this dataset lies less in the error rate and more in the fact that we can make good predictions for these patterns without resorting to perceptual accentuation rules.

5 Discussion

The model seems to perform basic meter categorization relatively well. It performed at competitive levels on both the Essen and the Finnish simulations. Furthermore it achieved good performance without risk of undergeneralizing due to overfitting from supervised learning. One area of current research is to see how well the model does at aligning (identifying the location of downbeats) in the Essen and Finnish databases.

As evidenced by the Povel & Essens results, the model has potential for performing alignment of an induced metrical hierarchy with a musical sequence. Though we have many other examples of this ability performance, including some entertaining automatic drumming to Mozart compositions, we have yet to undertake a methodical study of the the limitations of our model on alignment. This, and related tasks like online beat induction, are areas of ongoing research.

6 Conclusions

This paper introduces a novel way to detecting metrical structure in a music and to use meter as an aid in detecting tempo. Two main ideas were explored in this paper. First we discussed an improvement to using autocorrela-

tions for musical feature extraction via the computation of an autocorrelation phase matrix. We also discussed computing the Shannon entropy for each lag in this matrix as a means for sharpening the standard autocorrelation. Second we discussed ways to use the autocorrelation phase matrix to compute an alignment of a metrical hierarchy with music. We applied the model to the tasks of meter prediction and tempo induction on large datasets. We also provided preliminary results for aligning the metrical hierarchy with the piece (downbeat induction). Though much of this work is preliminary, we believe the results in this paper suggest that the approach warrants further investigation.

ACKNOWLEDGEMENTS

We would like to thank Fabien Gouyon, Petri Toiviainen and Tuomas Eerola for many helpful email correspondences.

References

J.C. Brown. Determination of meter of musical scores by autocorrelation. *Journal of the Acoustical Society of America*, 94:953–1957, 1993.

A. T. Cemgil and H. J. Kappen. Monte Carlo methods for tempo tracking and rhythm quantization. *Journal of Artificial Intelligence Research*, 18:45–81, 2003. URL `http://carol.science.uva.nl/~cemgil/papers/cemgil03a.pdf`.

A. T. Cemgil, H. J. Kappen, P. Desain, and H. Honing. On tempo tracking: Tempogram representation and Kalman filtering. *Journal of New Music Research*, 28:4: 259–273, 2001. URL `http://carol.science.uva.nl/~cemgil/papers/cemgil-tt.pdf`.

Grosvenor Cooper and Leonard B. Meyer. *The Rhythmic Structure of Music*. The Univ. of Chicago Press, 1960.

Simon E. Dixon. Automatic extraction of tempo and beat from expressive performances. *Journal of New Music Research*, 30(1):39–58, 2001. URL `http://www.oefai.at/~simon/pub/2001/jnmr.pdf`.

Douglas Eck. Finding downbeats with a relaxation oscillator. *Psychol. Research*, 66(1):18–25, 2002.

T. Eerola and P. Toiviainen. Digital Archive of Finnish Folktunes, 2004. [computer database]. University of Jyvaskyla. http://www.jyu.fi/musica/sks.

Masataka Goto. An audio-based real-time beat tracking system for music with or without drum-sounds. *Journal of New Music Research*, 30(2):159–171, 2001. URL `http://staff.aist.go.jp/m.goto/PAPER/JNMR2001goto.pdf`.

F. Gouyon, A. Klapuri, S. Dixon, M. Alonso, G. Tzanetakis, C. Uhle, and P. Cano. An experimental comparison of audio tempo induction algorithms, 2005. Submitted.

Stephen Handel. *Listening: An introduction to the perception of auditory events*. MIT Press, Cambridge, Mass., 1993.

A. Klapuri, A. Eronen, and J. Astola. Analysis of the meter of acoustic musical signals. *IEEE Trans. Speech and Audio Processing*, 2005. To appear.

Edward W. Large and J. F. Kolen. Resonance and the perception of musical meter. *Connection Science*, 6:177–208, 1994.

R. Parncutt. A perceptual model of pulse salience and metrical accent in musical rhythms. *Music Perception*, 11:409–464, 1994.

D.J Povel and Peter Essens. Perception of temporal patterns. *Music Perception*, 2:411–440, 1985.

H. Schaffrath. The Essen Folksong Collection in Kern Format, 1995. [computer database]. Center for Computer Assisted Research in the Humanitites.

E. Scheirer. Tempo and beat analysis of acoustic musical signals. *Journal of the Acoustical Society of America*, 103(1):588–601, 1998. URL `http://web.media.mit.edu/~eds/beat.pdf`.

Petri Toiviainen and Tuomas Eerola. The role of accent periodicities in meter induction: a classificatin study. In S.D. Lipscomb, R. Ashley, R.O. Gjerdingen, and P. Webster, editors, *The Proceedings of the Eighth International Conference on Music Perception and Cognition (ICMPC8)*, Adelaide, Australia, 2004. Causal Productions.

Anja Volk. Exploring the interaction of pulse layers regarding their influence on metrical accents. In S.D. Lipscomb, R. Ashley, R.O. Gjerdingen, and P. Webster, editors, *The Proceedings of the Eighth International Conference on Music Perception and Cognition (ICMPC8)*, Adelaide, Australia, 2004. Causal Productions.

P.G. Vos, A. van Dijk, and L Schomaker. Melodic cues for metre. *Perception*, 23:965–976, 1994.