MODELLING THE SPEED OF MUSIC USING FEATURES FROM HARMONIC/PERCUSSIVE SEPARATED AUDIO

Anders Elowsson **Anders Friberg**
KTH Royal Institute of Technology,
CSC, Dept. of Speech, Music and Hearing
elov@kth.se afriberg@kth.se

Guy Madison **Johan Paulin**
Department of Psychology, Umeå University
{Guy.Madison,Johan.Paulin}@psy.umu.se

ABSTRACT

One of the major parameters in music is the overall speed of a musical performance. In this study, a computational model of speed in music audio has been developed using a custom set of rhythmic features. Speed is often associated with tempo, but as shown in this study, factors such as note density (onsets per second) and spectral flux are important as well. The original audio was first separated into a harmonic part and a percussive part and the features were extracted separately from the different layers. In previous studies, listeners had rated the speed of 136 songs, and the ratings were used in a regression to evaluate the validity of the model as well as to find appropriate features. The final models, consisting of 5 or 8 features, were able to explain about 90% of the variation in the training set, with little or no degradation for the test set.

1. INTRODUCTION

This study is focused on one of the major parameters in music, the overall speed of a musical performance. From a music theoretic background we are used to associate speed with the tempo of the music. However, as suggested earlier, the perceived speed is related to the tempo but may also depend on other aspects like the note density (number of onsets per second) [9]. An indirect indication of this was provided in [2] where it was found that the note density (and not the tempo) was constant for a certain emotional expression across different music examples. Madison & Paulin [12] asked listeners to rate the speed for 50 music examples spanning a variety of musical styles and rhythms. They found that speed correlated with tempo but that there must also be other aspects involved in the perceptual judgment of speed. In earlier works [11, 15, 16] it has been shown that a classification of songs as fast or slow has helped to improve the accuracy of tempo estimation algorithms.

The current work is part of an ongoing study about perceptually determined features in music information retrieval. In a previous study it was shown that speed could be modeled by a combination of tempo and different note densities of the instruments using symbolic data [7]. The explained variation was about 90 % using linear regression. This indicates that a similar result could in theory be obtained using audio data provided that the appropriate low-level audio features could be extracted. Unfortunately, audio features extracted with the MIRToolbox [14] as well as the VAMP plugins available in the Sonic Annotator[1] did not map well to the perception of speed, highlighting the need for new features to be developed [7].

The purpose of the current study was to develop a computational model of speed in music audio restricted to examples containing percussive elements (e.g. drums). A set of rhythmic features were computed, mainly from detected onsets of the music. An important idea was that a relevant model should exploit the characteristics of each onset to better understand the music. As indicated in [7], good results can be achieved by tracking both percussive and harmonic onsets. Therefore, these parts were analyzed separately in the current model. As a first step, source separation was used to separate harmonic content and percussive content in the audio. Onsets and features were computed from both the percussive and the harmonic part as well as from the original audio. A flowchart of the processes used is shown in Figure 1.

To find appropriate features as well to evaluate the validity of the model, regression was used, in which the audio features were mapped against ground truth data consisting of listener ratings of speed.

2. SOURCE SEPARATION AND ONSET DETECTION

2.1 HP-Separation

Source separation has been used in the past in computational models related to rhythm [1]. For this study, the source separation method proposed by FitzGerald [6] was used to separate harmonic and percussive content. The basic idea of the method is that percussive sounds are broadband noise signals with short duration and that harmonic sounds are narrow band signals with longer duration. The audio is first transformed to the spectral domain by using a short-time Fourier transform (STFT). By applying a median filter across each frame in the frequency direction, harmonic sounds are suppressed. By applying a median filter across each frequency bin in the time direction percussive sounds are suppressed. After median filtering, the signal is transformed back to the time domain again using the inverse STFT.

With the STFT it is possible to accurately detect percussive content in the music. The frequency resolution in

[1] http://www.omras2.org/SonicAnnotator

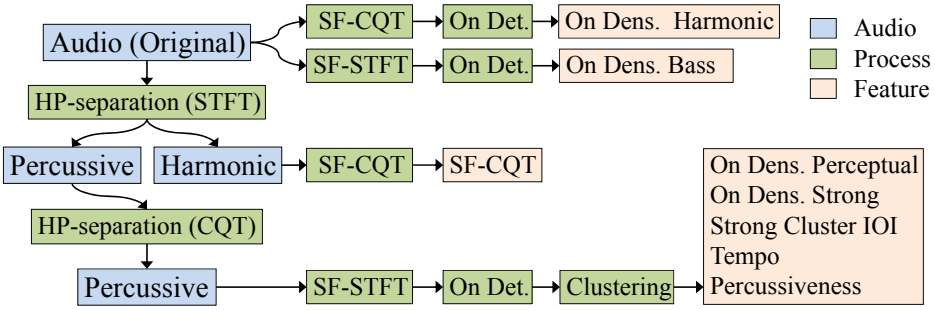

Figure 1. Flowchart of the processes used to compute audio features for the speed in music. The audio is filtered to separate harmonic and percussive content, onsets are detected from a spectral flux, and audio features are computed.

the lower frequencies is however not sufficient to detect harmonic content there. Thus, to further suppress harmonic content in the percussive waveform a second separation stage incorporates a constant-Q transform (CQT) [17]. The CQT can be understood as an STFT with logarithmically spaced frequency bins, accomplished by varying the length of the analysis window. With the CQT, an appropriate frequency resolution can be achieved at all frequencies, at the expense of a poor time resolution in the low frequencies. The frequency resolution of the CQT was set to 60 bins per octave and each frame was median filtered across the frequency direction. After filtering, the percussive signal was transformed back to the time domain using an inverse CQT. Notice that the phase information is retained in the transformation back to the time domain. It can be regarded as a mapping that connects a frequency bin to a certain point in time. The percussive and harmonic waveforms are shown in Figure 2.

2.2 Onset Detection

Audio features were computed from all three waveforms (original, harmonic and percussive) by the scheme shown in Figure 1. The first step, independent of feature and waveform, was to compute a spectral flux (SF) [3], where spectral fluctuations along the time-domain are detected. The SF was computed several times in numerous different ways. Some shared steps will be described here, with unique steps described in Sections 3.1-3.8. The power spectrum was computed with a CQT or a STFT and converted to sound level. A range of 30 dB was used. Thus, the maximum sound level of each band was set to 0 dB and sound levels below -30 dB were set to -30 dB. Let *L(n, i)* represent the sound level at the *i*th frequency bin/band of the *n*th frame. The SF is given by

$$SF(n) = \sum_{i=1}^{b} H\left(L(n,i) - (L(n-s,i)\right) \qquad (1)$$

where *b* is the number of bins/bands. The variable *s* is the step size and *H* is a half-wave rectifier function, or for the percussive SF:

$$H(x) = \begin{cases} x & \text{if } x > 0 \\ 0.2x & \text{if } x \le 0 \end{cases} \qquad (2)$$

The implication of Eq. 2 is that negative spectral fluctuations have a slight influence on the onset detection function. Onsets were detected by peak picking on a low-pass filtered curve of the spectral flux (see Figure 2).

2.3 Clustering

Onsets were clustered based on sound level in 8 frequency bands, spaced approximately an octave apart. An additional band was based on the RMS sound level. As the appropriate number of clusters was unknown beforehand, three K-means clusterings were carried out, with the number of clusters *k*, set to 2, 3 and 4. The fit of each clustering attempt was defined by the smallest Euclidian distance between any two clusters, where a large smallest distance gave a higher fit. When choosing *k*, a higher number of clusters were premiered over a lower if their fit was similar. The result of the clustering is a separation of onsets into different groups as shown in Figure 2.

3. FEATURE EXTRACTION

A total of 8 audio features were computed, 2 from the original waveform, 5 from the percussive waveform and 1 from the harmonic waveform. These features are shown in the flowchart in Figure 1 and described in Sections 3.1-3.8, with one subsection for each feature. An in-depth visualization of the processes involved to compute the features is shown in Figure 2. For conversion to *onset density*, the length of each song was set as the distance between the first and last onset.

3.1 Onset Density – Harmonic

Onsets were tracked from the original waveform, using the SF of a CQT. To avoid false onset detections at pitch glides, deviations in a peak by 20 cents (one bin), without an increase in sound level, were restricted from affecting the SF. This was accomplished by subtracting the sound level of each bin of the new frame, by the maximum sound level of the adjacent bins in the old frame.

3.2 Onset Density – Bass

To comply with the bass feature in [7], onsets in the low register (40 Hz - 210 Hz) were tracked using the SF of the lower bins of a STFT. The frequency bins were summed to a single band before the SF.

3.3 Onset Density – Perceptual weighting

Percussive onsets were tracked using the SF of a STFT on the percussive waveform. The bins of the frequency

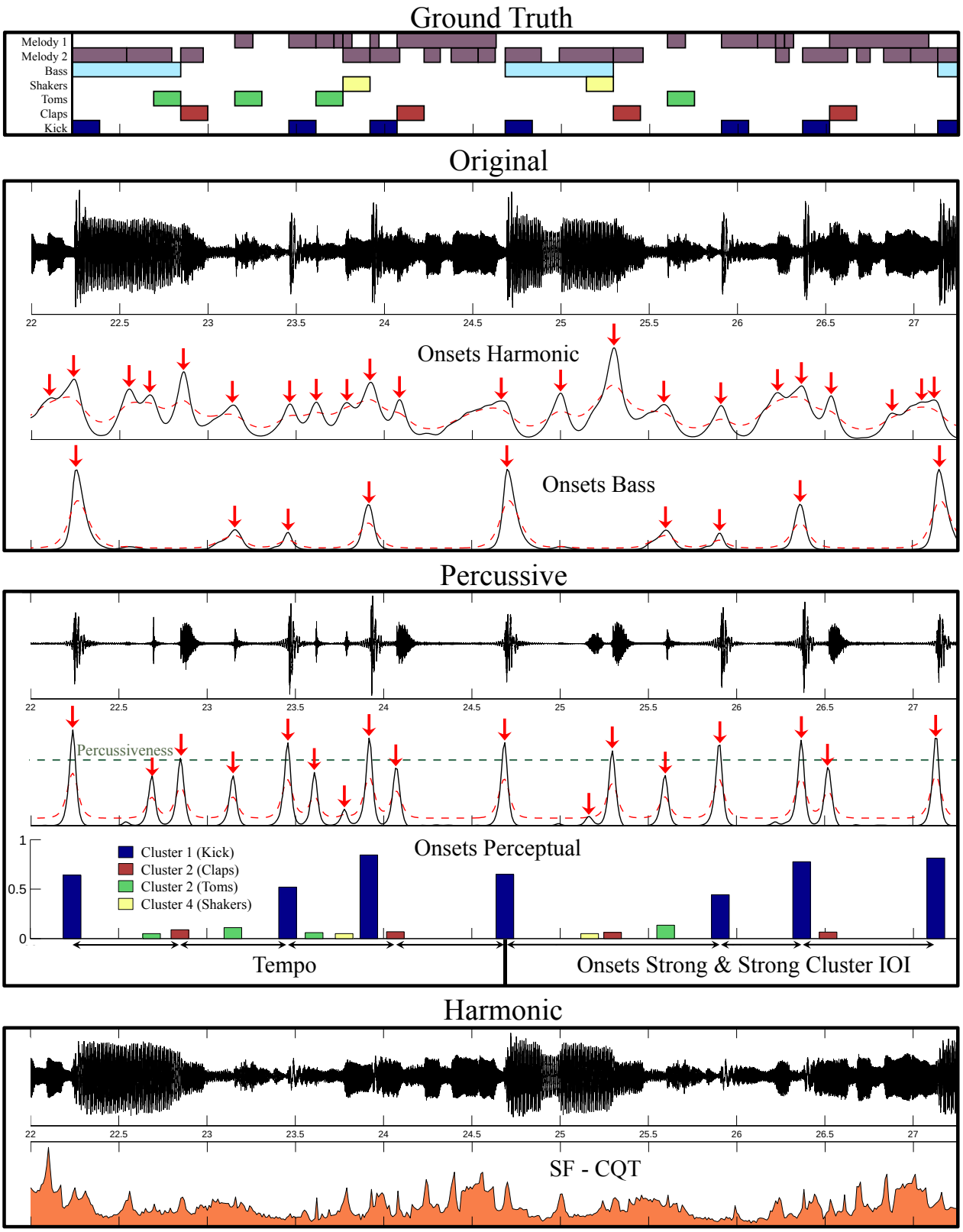

Figure 2. The process of estimating the perceived speed of a piece of music. The example is a 5-second section of the song *Candy Shop*, by *50 cent*. In the top pane we see the ground truth of the audio file. The melodic lines have been consolidated into a single row to convey only onset times. In the next pane we see the processes involved in extracting audio features from the original waveform. In the third pane the percussive audio is used. Notice that the clustering of the audio matches the ground truth. *Tempo* is detected as the IOI between kick and handclaps. Finally the integral of the spectral flux is used from the harmonic waveform in the fourth pane.

domain representation were divided into 13 non-overlapping frequency bands (half-octave spacing). Sub-band processing for onset detection has been described in [13], and can be motivated by its similarity to human hearing [4]. The strength of each detected onset was calculated based on the average sound level of the first

50 ms from the onset position, where lower frequencies were given a higher impact.

To further determine the perceived strength of the onsets, each onset was compared to the surrounding onsets within 1.5 seconds. This time span was defined as the perceptual present of the particular onset. By comparing it with the strongest onset within the perceptual present its strength could be altered to represent its perceptual impact. The onset was given a higher strength if there were no significantly stronger onsets within the perceptual present. If there were onsets that were significantly stronger, its strength was lowered. The height of the cluster-bars in Figure 2 represents the perceptual strength. To derive at a measure of onsets density, the sum of the perceptual strength of the onsets was used.

3.4 Onsets Density – Strong

The strongest clusters of the clustering contributed to two features. The first feature was simply the number of onsets, belonging to a strong cluster, per second. The idea behind this feature is that prominent percussive elements such as the kick drum and the snare drum likely influence the perception of speed in a different way than the less prominent elements such as the hi-hat.

3.5 Strong Cluster IOI

The second feature derived from the strong clusters was developed to catch the assumed perception of a *slow* speed, when the interonset intervals (IOIs) of onsets belonging to the same strong cluster are long. As an example, a song with equally spaced drum onsets consisting of "Kick, Snare, Kick, Snare, etc.." was assumed to have a *higher* perceived speed than a song where the drums instead plays "Kick, Kick, Snare, Kick, etc..". This is accounted for in the *Tempo* feature as well, because the tempo in the second example would be half the tempo of the first example. Cluster IOIs shorter than 750 ms were discarded based on the idea that they can both represent a drum fill in a slow song or represent a regular part of the drum pattern in a fast song.

3.6 Tempo

The tempo detection algorithm is part of an ongoing project, and a detailed description is in preparation. All distances between onsets within 5 seconds from each other are used to detect the tempo. The histogram in Figure 3 is based on the song presented in Figure 2.

First, the period length of the percussive waveform is detected. A histogram of onset distances is generated, where the contribution of each onset-pair is increased with increasing *similarity* in spectrum as well as increasing onset strength. The leftmost peak in the low pass filtered histogram, within 92 % of the highest peak, is chosen as the period length.

Secondly, the tempo (beat length) is detected. A histogram over onset distances is once again generated, where the contribution of each onset-pair is increased with increasing *dissimilarity* in spectrum as well as increasing onset strength. The final probability distribution for tempo (Figure 3) is the Hadamard product of the histogram and several filters. One filter is based on the determined period length. The idea is that the beat will be a simple ratio of the period length, so Hanning windows are produced at the positions given by

$$P_{len} \times \left(\frac{1}{2}\right)^n, \quad P_{len} \times \left(\frac{1}{2}\right)^n \times \left(\frac{1}{3}\right) \qquad n = 0,1,2,.. \tag{3}$$

Another filter is based on IOIs within strong clusters as described in Section 3.5 and several filters are based on onset density. The highest peak in the final probability distribution is chosen as the tempo. In compliance with the findings that speed is a shallower function of tempo for fast and slow music [12], differences in tempo between 60 and 160 BPM are given the highest impact.

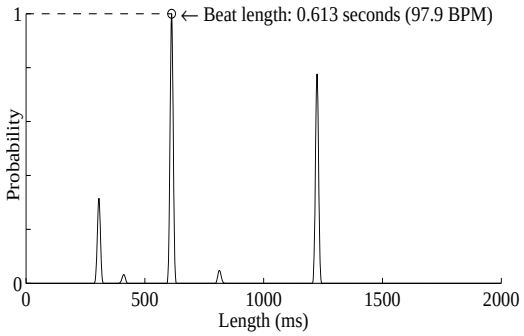

Figure 3. The histogram used to determine tempo.

3.7 Percussiveness

An estimate of how percussive the music is was computed as well. This estimate is derived from the height *h* of the peaks in the SF of the percussive waveform, as shown in Figure 2. Equation 4 gives the mean peak height when *p* is 0, an estimate closer to the lowest peaks when *p* is negative, and an estimate closer to the highest peaks when *p* is positive. In this study *p* was set to 0.4.

$$Percussiveness = \frac{\sum_{i=1}^{n} h(\mathrm{i})^{1+\mathrm{p}}}{\sum_{i=1}^{n} h(\mathrm{i})^{\mathrm{p}}} \tag{4}$$

3.8 SF CQT

When extracting information from the harmonic waveform the integral of the SF was used; indicated as the area in the bottom pane of Figure 2. Onset detection was avoided as the HP-separation had removed all transients from the harmonic waveform.

4. PREDICTING SPEED FROM THE FEATURES

4.1 Speed Data and Audio Examples

The music examples were taken from two earlier studies. To ensure that the songs contained percussive elements, songs where the RMS of the percussive waveform was less than 1/8 of the RMS of the harmonic waveform were not included in the data sets. The training set was 89 popular songs, originally in MIDI format and converted

to audio in a previous experiment [7, 10]. The speed estimations were previously determined using 20 listeners who rated speed for each example on a quasi-continuous scale marked slow-fast (range 1-9). The test set consisted of 47 real audio examples previously used for studying the relation between tempo and speed [12]. They were selected for exhibiting a large variation of tempi and genres within popular music styles. The speed was previously estimated in a similar way to the training set using continuous scales (range 0-10). Due to a difference in the design of the original experiment [12], the medium tempo examples were rated by 60 listeners while the fast and slow examples were rated by 12 listeners.

4.2 Modelling Speed of the Training Set

Two regression techniques were used to analyze the mapping between the computed audio features and the listener ratings. First, a multiple linear regression (MLR) was used, justified by a predictor-to-case ratio higher than 1:10. Secondly, partial least square regression (PLS) was used. PLS regression carries out data reduction, whilst maximizing covariance between features and predicted data [5]. It constructs new predictor variables (components), as linear combinations of the features.

The MLR prediction of listener ratings from computed audio features is presented in Table 1. As shown, a linear combination of the computed audio features was able to explain more than 90 % of the variability. In comparison, the agreement among the listeners, estimated by the mean intersubject correlation was 0.71 and Cronbach's alpha 0.98 [7].

8 Features	**R^2 = 0.909**	**Adjusted R^2 = 0.900**	
Variable	**beta**	***sr*²**	**p-value**
On Dens. - Harmonic	0.205	0.033	0.000***
On Dens. - Bass	0.130	0.007	0.016*
On Dens. - Perceptual	0.302	0.018	0.000***
On Dens. - Strong	-0.155	0.010	0.004**
Strong Cluster IOI	0.127	0.006	0.021*
Tempo	0.430	0.056	0.000***
Percussiveness	-0.095	0.005	0.041*
SF CQT	0.107	0.004	0.053
5 Features	**R^2 = 0.887**	**Adjusted R^2 = 0.880**	
On Dens. - Harmonic	0.239	0.049	0.000***
On Dens. - Perceptual	0.224	0.020	0.000***
Strong Cluster IOI	0.132	0.007	0.027*
Tempo	0.404	0.053	0.000***
SF CQT	0.225	0.032	0.000***

Table 1. MLR prediction of the perceptual feature *speed* from computed audio features. The variable sr^2 is the squared semipartial correlation coefficient.

For the model based on 8 features, 2 features (*Onset Density – Strong* and *Percussiveness*) gave a negative contribution. Notice that the difference in explained variance is only about 2 % between the two models, indicating that the features in the 5-feature model may contain almost all relevant information.

The PLS regression of the 8 features is shown in Table 2. With 3 components, the cross-validated adjusted R^2 indicates that just below 90 % of the variability could be explained. Note also that the cross-validation procedure only lowers the result marginally, supporting the validity of the present features.

PLS Regression – Speed (3 PLS-components)		
R^2 = 0.907	**Adj. R^2 = 0.903**	**Adj. R^2 cv = 0.878**
Component	**Explained variance**	**Cum. variance**
1	0.845	0.845
2	0.052	0.897
3	0.017	0.914

Table 2. PLS prediction of the perceptual feature *speed* from computed audio features. The squared correlation coefficient R^2 was derived using PLS, including 10-fold cross validation ("cv"). Also, R^2 as a function of the number of components is shown.

The fitted values of the linear regression from Table 1 (8-feature model) are shown in Figure 4 below. As seen in the Figure, the deviations from the target are rather evenly distributed across the range and with a maximal deviation of about one unit.

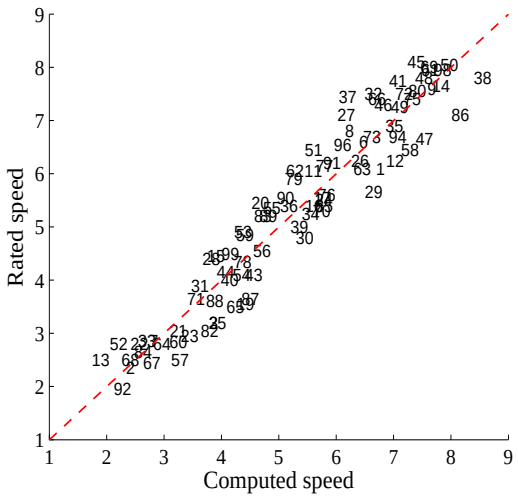

Figure 4. The fitted values in the MLR prediction of perceptual speed, where higher means faster. For each song, the x-axis represents the estimated speed and the y-axis represents the ground truth (derived from listeners).

4.3 Predicting Speed of the Test Set

Two linear models of speed (5 and 8 features) were derived from the multiple linear regression analysis of the training set shown in Table 1. The models were applied to the test set and the squared correlation between rated speed and computed speed is shown in Table 3.

No. of Features/Regression coefficients	**R^2**
5	0.934
8	0.894

Table 3. The prediction of the perceptual feature *speed* from a linear model using computed audio features.

The 5-feature model's prediction of speed for each song of the test set is shown in Figure 5. Computed speed

is approximately 1 unit higher than rated speed and this is probably due to the differences in the music examples of the databases. Furthermore, different scales were used for the listener ratings in the two data sets (1-9 and 0-10).

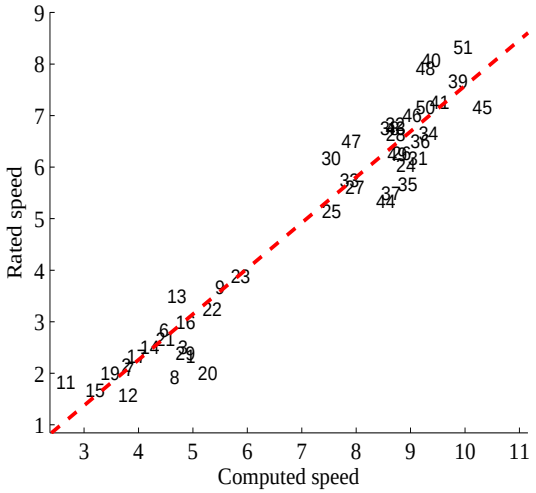

Figure 5. The prediction of the perceptual feature speed, where higher means faster. For each song, the x-axis represents the estimated speed and the y-axis represents the ground truth (derived from listeners).

5. CONCLUSIONS AND DISCUSSION

The models were able to explain about 90 % of the variability in listener ratings. The most important features were tempo together with onset densities for different layers of the music as well as spectral fluctuations in the harmonic part of the audio. The validity of the features was supported by cross-validation, and verified by using the extracted regression coefficients from the training set to accurately predict speed in the test set.

The results show that it was possible to reach the same high explained variance on audio data as on the symbolic data in [7] using similar features. This indicates that the appropriate low-level audio features have been extracted, which is reassuring for the ongoing study. The model based on 5 features was able to explain more of the variance in the test set than the model based on 8 features. This indicates that the 8-feature model was overfitting the training set.

The segmentation of audio (HP-separation and clustering) seems to be a promising path forward. By clustering onsets we can detect onsets belonging to the same source and thus use the rhythmic pattern of this source in the model. By using several onset detection functions on separate parts of the audio, different aspects of the music can be captured. Source separation can be motivated from an ecological perspective; it seems reasonable to assume that listeners distinguish between sounds from different sources to better understand the soundscape. A drawback with the proposed system is that the computation of several STFTs and CQTs is relatively time consuming.

In future work we intend to include songs without percussive elements. We also intend to investigate other high level rhythmic features such as rhythmic complexity and dynamics. We expect the audio segmentation to be a fruitful way forward. Data from this study is freely available for research purposes[2].

6. ACKNOWLEDGEMENT

This work was supported by the Swedish Research Council, Grant Nr. 2009-4285 and 2012-4685.

7. REFERENCES

[1] M. Alonso, G. Richard and B. David: "Accurate Tempo Estimation Based on Harmonic + Noise Decomposition," *Journal on Advances in Signal Processing*, 2007.

[2] R. Bresin, and A. Friberg: "Emotion Rendering in Music: Range and Characteristic Values of Seven Musical Variables," *Cortex*, Vol. 47, No. 9, pp. 1068-1081, 2011.

[3] S. Dixon: "Onset detection revisited," In *Proc. of DAFx*, pp. 133–137, 2006.

[4] C. Duxbury, J. P. Bello, M. Sandler, and M. Davies: "A Comparison between Fixed and Multiresolution Analysis for Onset Detection in Musical Signals," In *Proc. of DAFx,* pp. 207-212, 2004.

[5] T. Eerola, O. Lartillot, P. Toiviainen: "Prediction of Multidimensional Emotional Ratings in Music from Audio using Multivariate Regression Models," In *Proc. of ISMIR,* pp. 621-626, 2009.

[6] D. FitzGerald: "Harmonic/Percussive Separation Using Median Filtering," In *Proc. of DAFx,* 2010.

[7] A. Friberg, E. Schoonderwaldt, A. Hedblad, M. Fabiani, and A. Elowsson: "Perceptually derived features can be used in music information retrieval," submitted.

[8] A. Friberg, E. Schoonderwaldt, and A. Hedblad: "Perceptual Ratings of Musical Parameters," In von H. Loesch, and S. Weinzierl (Eds.), *Gemessene Interpretation - Computergestützte Aufführungsanalyse im Kreuzverhör der Disziplinen,* pp. 237-253, Mainz: Schott, 2011.

[9] A. Gabrielsson: *Studies in rhythm*, doctoral dissertation, Uppsala University, 1973.

[10] A. Hedblad: *Evaluation of musical feature extraction tools using perceptual ratings*. Master thesis, KTH Royal Institute of Technology, 2011.

[11] J. Hockman and I. Fujinaga: Fast vs Slow: Learning Tempo Octaves from User Data. In *Proc. of ISMIR*, pp. 231-236, 2010.

[12] G. Madison, and J. Paulin: "Ratings of Speed in Real Music as a Function of both Original and Manipulated Beat Tempo." *Journal of the Acoustical Society of America*, Vol. 128, No. 5, pp. 3032-3040, 2010.

[13] A. Klapuri: "Sound Onset Detection by Applying Psychoacoustic Knowledge," In *Proc. IEEE Conf. Acoustics, Speech and Signal Processing*, 1999.

[14] O. Lartillot and P. Toiviainen: A Matlab Toolbox for Musical Feature Extraction from Audio. In *Proc. of DAFx*, pp. 237-244, 2007.

[15] M. Levy: Improving Perceptual Tempo Estimation With Crowd-Sourced Annotations. In *Proc. of ISMIR*, pp. 317-322, 2011.

[16] G. Peeters and J. Flocon-Cholet: Perceptual Tempo Estimation Using GMM Regression. In *Proc. of ACM MIRUM*, pp. 45-50, Japan, November 2012.

[17] C. Schörkhuber and A. Klapuri: "Constant-Q Transform Toolbox for Music Processing," In 7th Sound and Music Conference, Barcelona, 2010.

[2] www.speech.kth.se/music/speed