WEARABLE ASSISTANCE FOR THE BALLROOM-DANCE HOBBYIST – HOLISTIC RHYTHM ANALYSIS AND DANCE-STYLE CLASSIFICATION

Florian Eyben, Björn Schuller, Stephan Reiter, and Gerhard Rigoll

Institute for Human-Machine Communication
Technische Universität München, Germany
[eyb, sch, res, rl]@mmk.ei.tum.de

ABSTRACT

Automated retrieval of high level information from ballroom dance music is challenging, but has many practical applications. These include, for example, a fully automatic ballroom dance D.J., robots capable of performing ballroom dances, or wearable dance-assistance, as considered herein. It is necessary, for such a system, to retrieve information about the song’s quarter note tempo, meter and beat positions. Further, the system must be able to discriminate between the nine Standard and Latin ballroom dances. In this paper we present a model that combines all these requirements in one holistic approach. The polyphonic input is processed by a simplified psychoacoustic model. Tatum, tempo and meter features are extracted using resonant filters. The filter output is used for beat tracking. The extracted features are used for a ballroom dance-style classification by Support-Vector-Machines. To show the high effectiveness regarding dance-style recognition and beat tracking, test-runs are carried out on a database containing 1.8k titles.

1. INTRODUCTION

Especially for the ballroom dance beginner or hobbyist a wearable ballroom dance assistance would be beneficial. Such a device should be able to recognize the ballroom dance-style, quarter note tempo and meter. Further, beat positions and the beginning of a meter must be identified by the assistant, in order to give the dancer hints about the current dance-style, the counting, i.e. when to start, or which dance steps the dancer has to perform on a certain beat. These hints can be delivered, for example, via an earplug from an individual wearable computing device or as a public announcement through the ballroom speaker system.

Tempo and meter detection is the basis of such a ballroom dance assistant. Therefore, a few existing tempo detection and meter recognition approaches [2-5] are briefly summarized in section 2, before we present our model that combines both capabilities in one model. As described in section 3 in more detail, the model uses comb filters as resonators [4] to detect tempi on several metrical levels, extract features for ballroom dance-style classification and perform beat tracking using the comb filter output. Test-runs are carried out on a 1.8k public database [1], introduced in section 4. Ballroom dance-style classification is discussed in section 5. Finally, results and conclusion are found in section 6 and 7, respectively.

2. GENERAL APPROACH

Information about musical meter is essential for ballroom dance-style recognition. By using only the meter, it can be distinguished between Waltzes (triple meter) and the other seven dance-styles (duple meter or common time). Meter detection requires tempo independent information about the rhythmic structure of the audio track [3]. Thus, it is necessary to reliably detect the tempo of the song in beats-per-minute. There are basically three different methods that are commonly used for tempo detection: correlation methods, note onset detection followed by computation of the most common inter-onset-interval (IOI) via a histogram method [5] and a multiple resonator based approach using IIR comb filters [4]. Our tempo detection method is based on [4], except for a few improvements and performance enhancements. As it is very difficult, even for a human listener, to determine if the quarter notes of a song are better grouped by 2, 4 or 8, we restrict the meter recognition problem to a simple duple or triple decision, as done in [2]. This is sufficient for ballroom dance music. The algorithms mentioned in [2] and [5] try to determine the metrical grouping by explicitly identifying downbeats. A downbeat thereby is a stronger accented beat, which might indicate the first beat in a meter. However, the task of reliably finding a downbeat is challenging, even for the non-experienced human listener. It must also be considered that downbeats need not always occur at the beginning of a meter. We therefore focus on a reliable two-class decision between duple and triple meter without any knowledge about note onset or downbeat positions. Moreover, other approaches to meter detection (such as [2]) require manual adjustments of the automatically extracted quarter note tempo. Our approach is able to discriminate between duple and triple meter without knowledge of the quarter note tempo. Instead, the fastest tempo present in the song, called the tatum tempo, is detected. Then, the distribution of resonances across 19 fractions of this tatum tempo is analyzed. This distribution combined with information from the tatum detection stage reveals highly relevant information about the metrical structure, that we use for quarter note tempo detection and ballroom dance-style classification.

Once the quarter note tempo, the meter and the ballroom dance-style are known, finding beat positions and downbeat locations is made much easier. With the assumption that a track of typical ballroom dance music has a roughly constant tempo throughout, the most likely phase for a grid of nearly equidistant beats is determined for small excerpts of the song by analyzing the output of a comb filter tuned to the detected quarter note tempo. For deciding which of the labeled beats is a downbeat, loudness information from the psychoacoustic model is used.
3. FEATURE EXTRACTION

In the process of extracting the features tempo and meter, several other features are extracted, that will be used in the later classification step. Before any feature extraction can be performed a perceptive preprocessing using a simplified psychoacoustic model is applied to the polyphonic PCM input data.

3.1. Perceptive preprocessing

Input audio data is reduced to a sample rate of 11.025 kHz and converted into monophonic by stereo-channel addition in order to reduce computation time. Audio is then split into frames of 256 samples with an overlap of 0.57 so the resulting envelope frame-rate is 100 frames-per-second (fps). A half-wave sine window is used as windowing function when computing the FFT on each frame. By using \( N_{fee} \) overlapping triangular filters, equidistant on the Mel-Frequency scale, the 128 bands are reduced to \( N_{fee} \) nonlinear bands. Tests were carried out for different numbers for \( N_{fee} \) and the best overall result was achieved with \( N_{fee}=9 \). The envelope of each of the \( N_{fee} \) bands is then low pass filtered by convolving with a half-wave raised cosine filter having a length of 15 envelope samples [4]. This preserves fast attacks, but filters noise and rapid modulation, most as in the human ear. From the filtered band envelopes a weighted differential \( d_e \) is computed in the following way for a sample \( o \) at position \( i \): A moving average is calculated over one window of 10 samples to the left of sample \( o \), (left mean \( o_{i-1} \)) and a second window of 20 samples to the right of sample \( o \), (right mean \( o_{i+1} \)). The differential then is given by (1):

\[
d_e(i)=|o_i-\bar{o}_{i-1}|-|\bar{o}_{i+1}-o_i|
\]  

Taking into account data before and after the current position was inspired by [5]. The sound level before an onset event and the duration of the note belonging to the onset both contribute to the perceived emphasis of the specific onset event, which corresponds to the perceived accentuation of the note.

3.2. Tatum features

The tatum grid, according to [5], is the lowest metrical level of a song. It corresponds to the fastest tempo in the song, or likewise the smallest inter-onset-interval (abbr. IOI in the ongoing). For detecting the tatum tempo we use a comb filter bank consisting of 57 filters. We assign a fixed gain \( \alpha \) to each filter in the bank instead of deriving a gain for each filter basing on a constant half energy time as is done in [4]. Overall best results were achieved with a gain of 0.7. The delays of the filters range from \( r=18 \) to \( r=74 \) envelope samples. The differential \( d_e \) of each Mel-frequency-band envelope is processed by the comb filter bank and the total energy over all bands is a score value \( m \), which is stored in the meter vector at position \( i \). This score value \( m \) indicates how well the tempo \( T_{r,i} \) resonates with the song, i.e. how strong \( i \) is present compared to the other tempi. It is sufficient to use \( i \in [1; 19] \), since all important multiples for meter classification are included in this range. At this point we present a simple rule based method for discriminating between duplet and triple meter. Later on we will use Support-Vector-Machines to improve meter recognition accuracies. Two sums \( s_1 \) and \( s_2 \) are computed (\( m \) refers to the meter vector):

\[
s_1=\frac{1}{3}(m(4)+m(8)+m(16))
\]

\[
s_2=\frac{1}{3}(m(6)+m(9)+m(18))
\]

The greater of the two sums determines what we call base meter \( M \). If \( s_1 \) is greater, then \( M \) is triple (e.g. 3/4 or 6/8 time signature), otherwise it is duplet (e.g. 4/4 or 2/4 time signature).

3.3. Meter and Tempo Features

A meter vector \( m \) is computed by setting up small comb filter banks with filter delays \( \tau \) ranging from \( T_{r,i-1} \) to \( T_{r,i+1} \). For each filter bank with the highest energy output is selected as the tempo \( T_{r,i} \) and the total energy of this filter summed over all bands is a score value \( m \), which is stored in the meter vector at position \( i \). This score value \( m \) indicates how well the tempo \( T_{r,i} \) resonates with the song, i.e. how strong \( i \) is present compared to the other tempi. It is sufficient to use \( i \in [1; 19] \), since all important multiples for meter classification are included in this range. At this point we present a simple rule based method for discriminating between duplet and triple meter. Later on we will use Support-Vector-Machines to improve meter recognition accuracies. Two sums \( s_1 \) and \( s_2 \) are computed (\( m \) refers to the meter vector):

\[
s_1=\frac{1}{3}(m(4)+m(8)+m(16))
\]

\[
s_2=\frac{1}{3}(m(6)+m(9)+m(18))
\]
We also use the meter vector to identify the quarter note (main) tempo. Bounds within which the tempo is expected are defined. For dupel $M_d$, the range is from 60 BPM to 222 BPM and from 75 BPM to 240 BPM for triple $M_t$. For each $\theta$, within bounds that satisfies the condition $m_{32} > m_{30}$ for dupel $M_d$ or $m_{31} > m_{30}$ for triple $M_t$, a score $\sigma_i = m_{51} + m_{32} > m_{30}$ or $\sigma_i = m_{51} + m_{31}$, respectively, is computed and the $\theta$, for which $\sigma_i$ is maximal is chosen as main tempo $\theta_0$. If no $\theta$, within bounds is found that satisfies these conditions, the first within the defined tempo bounds having the highest score $m_i$ is chosen as main tempo $\theta_0$. Should it happen, that no $\theta$, falls within bounds, the tatum tempo $T_i$ is chosen for the main tempo $\theta_0$.

3.4. Beat tracking

A tracking envelope $E$, is computed by filtering $d_{n}$ of each Mel-Frequency band (see section 3.1.) by a comb filter with $\alpha=0.4$ and delay $\tau=\theta_0$ and adding the filter outputs over all $N_{mel}$ bands. Multiple pass beat tracking is performed using this tracking envelope. Beats may differ from their expected position by a tolerance $t$. Starting at frame 0 as the first beat candidate an iterative tracking with a large tolerance window is performed to find possible beat candidates. If $p_{n}$ is the position of the current beat candidate $n$, the position of the candidate $n+1$ is determined as the position of the highest peak in a window of $E_{i}$ between positions $p_{n}+t_{n}+t$ and $p_{n}+t_{n}+t$. The same tracking algorithm is now applied for each candidate $n$ as the first beat with a now smaller tolerance $t$ of 8 frames in order to find a reliable first beat and the optimal set of beat positions. This set is chosen by determining for which candidate the sum over the values of $E$, at the tracked beat positions is maximal, because the higher the amplitude of a peak in $E$ is, the more likely this peak corresponds to a beat. After a reliable first beat has been detected by analyzing at least 15 seconds of recorded audio, the beat tracking algorithm is capable of predicting beats on-line in real-time.

4. BALLROOM DANCE DATABASE

We chose a set of 1,855 pieces of typical Standard and Latin dance music publicly accessible at [1], covering the Standard Dances (Waltz, Viennese Waltz, Tango, Quick Step, and Foxtrot, and the Latin Dances Rumba, Cha Cha, Samba and Jive. 30 seconds of each song are available, which we converted to 44.1kHz PCM as required by our preprocessing. The distribution among dance-styles is depicted in tab. 1. The set is abbreviated BRD in the ongoing.

<table>
<thead>
<tr>
<th>Waltz</th>
<th>Viennese Waltz</th>
<th>Tango</th>
<th>Quick Step</th>
</tr>
</thead>
<tbody>
<tr>
<td>293</td>
<td>136</td>
<td>185</td>
<td>242</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Foxtrot</th>
<th>Rumba</th>
<th>Samba</th>
<th>Cha Cha</th>
<th>Jive</th>
</tr>
</thead>
<tbody>
<tr>
<td>245</td>
<td>217</td>
<td>188</td>
<td>211</td>
<td>138</td>
</tr>
</tbody>
</table>

Tab. 1: Distribution of dance-styles within BRD data-set

The ground truth of tempo, ballroom dance-style and meter for the BRD data-set is known from [1]. A part of this set is also used in [7] for ballroom dance-style recognition.

5. CLASSIFICATION

With all 83 features introduced in section 3 we now aim at classifying meter and ballroom dance-style. As classifier we employ Support Vector Machines (SVM) with a polynomial Kernel function basing on our experience in Musical Genre Discrimination [6]. We firstly analyze the dataset by performing a closed-loop Hill-climbing feature selection employing the target classifier’s error rate as optimization criterion, namely Sequential Forward Floating Search (SVM-SFFS) [7]. This reveals the following feature-set to yield the best results for ballroom dance-style classification ($FS_0$ in the ongoing): Base meter $M_0$, $T$-ratio, $T$-slope, $T$-peakdist, meter vector $m$ elements 4, 5, 6, 8, 11, 12, 14, 15, 19 and tatum vector $T$ elements 1-20, 22-28, 30-57. Features relevant for meter classification ($FS_0$ in the ongoing) are $T$-ratio, meter vector $m$ elements 4, 6, 8, 16 and tatum vector $T$ elements 3, 4, 7, 13, 16, 19, 20, 22, 23, 25, 26, 28, 29, 33, 36, 37, 39, 48, 50.

6. RESULTS AND DISCUSSION

Test runs have been carried out on the formerly introduced set BRD. The accuracy of the features tempo $\theta_0$ and base meter $M_0$, compared to the given ground truth of tempo and meter is evaluated. Hereby the tolerance for tempo detection is 2.5 frames absolute IOI deviation. It makes more sense to express the tolerance as allowed deviation of the IOI in frames than as relative BPM deviation, since slight tempo detection inaccuracies are due to roundoff errors because a frame-rate only 100fps is used. However, a quite sufficient accuracy for our purpose is achieved with this frame-rate and computation time is kept at a minimum, considering low CPU resources on a wearable computing device. When the detected tempo is twice or half the ground truth tempo (octave error), it is still treated as correctly detected, since the correct tempo octave can be determined using knowledge about tempo ranges of nine typical ballroom dance-styles.

Additionally, we evaluate the effectiveness for ballroom dance-style and meter classification by Support-Vector-Machines in a 10-fold stratified cross validation (SCV). Thereby evaluation splits are always disjunctive in terms of musical pieces. The data is standardized to have zero mean and unit variance with respect to the training split.
Using a P4-mobile with 1.4 Ghz, average computation time for the feature extraction is 1.3s for the 30s excerpts from the BRD set, roughly corresponding to a rtf of 0.05. This makes it possible to run the algorithm on a wearable computing device, e.g. a smartphone, in real-time.

Tab. 2 lists detailed per-dance-style results for quarter note tempo and meter recognition. Comparing these results with results from other work is not possible because other data-sets are used there. However, accuracies of up to 98% for certain dance-styles and mean accuracies around 90% definitely prove that our approach is state-of-the-art.

Beat tracking was evaluated on a subset 225 songs from the BRD set, containing 25 songs of each dance-style. Detailed results are also given in tab. 2. The results were evaluated considering three different aspects. (A) refers to the percentage of songs where beats were labeled correctly on at least one metrical level. Songs with tempo detection octave errors are included in this evaluation. (B) refers to the percentage of songs where beats on the quarter note level were labeled correctly. Using information about ballroom dance-style from the following classification-step the numbers for (B) could be improved by far. The relative percentage of identified downbeats indicates how many songs of those where the quarter note tempo was tracked correctly, the beginning of a meter (the “1”) was identified correctly. Due to the off-beat structure of Jive pieces half of them were tracked correctly regarding tempo, but at an off-beat phase. Thus, they were counted as incorrectly tracked.

By using data-driven classification with SVM (feature-set FSs), meter recognition accuracies (see tab. 3), are improved by 4%. Detailed results for Recall, Precision and F1-Measure for data-driven ballroom dance-style classification by SVM using feature-set FSs are printed in tab. 2. A total of 90% correctly classified instances can be reported for the latter.

Tab. 3. - Meter classif. by SVM, feature-set FSs, 10-fold SCV.

### Accuracy [%]

<table>
<thead>
<tr>
<th>Tempo ( \theta_t )</th>
<th>Jive</th>
<th>Samba</th>
<th>Rumba</th>
<th>Cha</th>
<th>Fox</th>
<th>QuickS</th>
<th>Waltz</th>
<th>VWaltz</th>
<th>Tango</th>
<th>MEAN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Base meter ( M_b )</td>
<td>100.0</td>
<td>96.8</td>
<td>99.5</td>
<td>94.3</td>
<td>95.0</td>
<td>73.7</td>
<td>75.7</td>
<td>97.3</td>
<td>90.3</td>
<td>93.3</td>
</tr>
<tr>
<td>Tempo &amp; base meter</td>
<td>98.6</td>
<td>93.6</td>
<td>89.9</td>
<td>97.6</td>
<td>90.2</td>
<td>89.7</td>
<td>69.6</td>
<td>75.7</td>
<td>93.0</td>
<td>87.8</td>
</tr>
<tr>
<td>Tracked beats (A)</td>
<td>52.0</td>
<td>77.0</td>
<td>85.0</td>
<td>89.9</td>
<td>84.0</td>
<td>72.0</td>
<td>68.0</td>
<td>84.0</td>
<td>96.0</td>
<td>77.3</td>
</tr>
<tr>
<td>Tracked beats (B)</td>
<td>28.0</td>
<td>70.0</td>
<td>77.0</td>
<td>81.0</td>
<td>84.0</td>
<td>72.0</td>
<td>68.0</td>
<td>84.0</td>
<td>52.0</td>
<td>62.2</td>
</tr>
<tr>
<td>Downbeats (rel. %)</td>
<td>14.3</td>
<td>52.6</td>
<td>63.8</td>
<td>40.0</td>
<td>61.9</td>
<td>50.0</td>
<td>85.2</td>
<td>38.5</td>
<td>62.9</td>
<td></td>
</tr>
</tbody>
</table>

### BDS Recall

| BDS Precision  | 92.8  | 81.9  | 78.8  | 85.8 | 94.7  | 92.1   | 94.9  | 91.9   | 89.8  |
| BDS F1-Measure | 92.4  | 86.5  | 78.6  | 89.2 | 95.5  | 89.9   | 92.1  | 89.2   | 89.9  |

**Tab. 2.** - Feature extraction accuracy for tempo and meter (Tempo tolerance is IOI±2.5 frames). Beat tracking accuracies (see text for details). Recall, Precision and F1-Measure of ballroom dance-style classification (BDS) with feature-set FSs.

8. REFERENCES


