DANCE MUSIC CLASSIFICATION: A TEMPO-BASED APPROACH

Fabien Gouyon
Universitat Pompeu Fabra
Barcelona, Spain

Simon Dixon
Austrian Research Institute for AI
Vienna, Austria

ABSTRACT

Recent research has studied the relevance of various features for automatic genre classification, showing the particular importance of tempo in dance music classification. We complement this work by considering a domain-specific learning methodology, where the computed tempo is used to select an expert classifier which has been specialised on its own tempo range. This enables the all-class learning task to be reduced to a set of two- and three-class learning tasks. Current results are around 70% classification accuracy (8 ballroom dance music classes, 698 instances, baseline 15.9%).

1. INTRODUCTION

Tempo is a musical attribute of prime importance. Moreover, recent research [3] advocated its relevance in the task of classifying different styles of dance music: focusing *solely* on the *correct* tempo (i.e. measured ***manually***) 8 classes of Standard and Latin ballroom dance music can be classified, by means of diverse classification techniques, with around 80% accuracy (total of 698 instances, classes are listed in Table 1, baseline is 15.9%). Decision tree classifiers revealed a clear ordering of dance styles with respect to tempi. Therefore, one can assume that, given a musical genre, the tempo of any instance is among a very limited set of possible tempi. For instance, the tempo of a Cha Cha is usually between 116 and 128 BPM. Table 1 gives tempo ranges for the 8 dance styles used here.

This assumption may be arguable, yet it seems to make sense for ballroom dance music as, on the one hand, common musical knowledge (e.g. instructional books, dance class websites [1]) suggests such boundaries, and on the other hand, [4] shows on a large amount of data (more than 90000 instances) that different dance music styles ("trance, afro-american, house and fast") show clearly different tempo distributions, centered around different "typical" tempi.

[1] see e.g. *http://www.ballroomdancers.com/Dances*

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page.
© 2004 Universitat Pompeu Fabra.

In this paper we propose to further exploit the high relevance of the tempo in designing a classifier that focuses first on this feature and then uses complementary features to make decisions in possibly ambiguous situations (i.e. tempo overlaps).

Our approach is to define the tempo range of each class by a Gaussian probability function. An illustration is given in Figure 1. The Gaussian standard deviations are defined so that the probabilities at the limits specified in Table 1 are half the value of the corresponding probability maximum. Put together, these probabilities may overlap in certain tempo regions (e.g. Samba and Rumba, see dashed-blue and solid-black lines around 100 BPM in Figure 1).

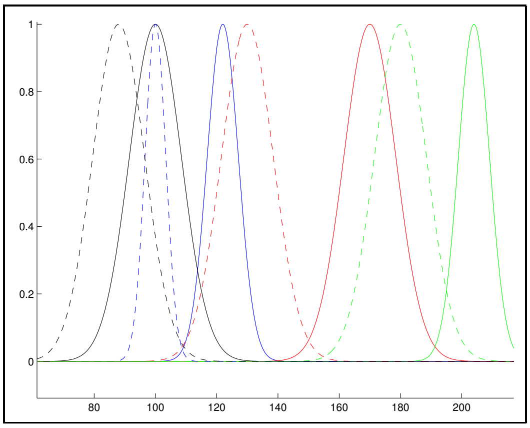

Figure 1. Tempo probability functions and overlaps of 8 dance music styles. X-axis in BPM.

Hence, given an unknown instance, a simple classification process could be:

1: Compute its tempo T
2: Retrieve the n classes that overlap significantly at T
3: Use a classifier tailored to these n classes

Assuming that different classes have different tempo distributions, it is reasonable to consider that much less than 8 classes do overlap *significantly* at any tempo. The classifier design is consequently easier than the eight-class learning task considering all examples.

However, there is a consensus in the tempo-tracking literature on the fact that state-of-the-art tempo induction algorithms typically make errors of metrical levels (they output e.g. half the correct tempo). Accordingly, the tempo-tracking algorithm we use here —BeatRoot [1]— may output the correct tempo, twice or half of its value (in the case of excerpts with a duple meter) or two thirds of its

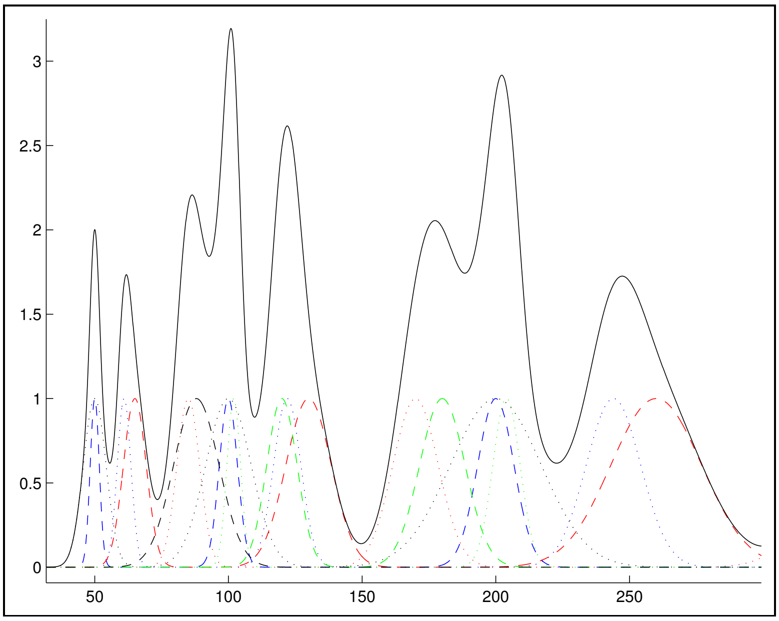

Figure 2. Adapted tempo probability functions of 8 dance music styles. X-axis in BPM. Solid black line is the sum of all probability functions and represents overall class overlaps.

value (in the case of excerpts with a triple meter).[2] We adapted each tempo probability function accordingly in concatenating several Gaussians whose means are correct tempi and relevant multiples (see Figure 2).

Observing the probability functions in Figure 2, one can see that each tempo value corresponds usually to two different potential classes, at the exception of three specific tempo regions in which three classes overlap. These are 95 to 105 BPM and 193 to 209 BPM for Quickstep, Rumba and Samba, and 117 to 127 BPM for Cha Cha, Tango and Viennese Waltz. Therefore, we propose to build 30 different classifiers:

- $\sum_{n=1}^{8-1} n = 28$ two-class classifiers, $\{K_1 \ldots K_{28}\}$, each expert in a specific pairwise classification task.
- 2 three-class classifiers, K_{29} and K_{30}, each expert in a three-class specific task

When presented with unknown instances, the knowledge available to the system is this set of 30 expert classifiers and the tempo probability functions for all possible classes. Therefore, the overall classification process is finally that detailed in Algorithm 1.

Algorithm 1 Overall classification process

1: Compute tempo T of the instance to classify
2: Find the classifier K_i whose tempo range includes T
3: Perform classification with K_i

In the remainder of this paper, we give the detail of the audio data used for experiments. Then we introduce the diverse features computed from this data. We then detail our experiments, discuss the results, compare them to the results reported in [3] and propose a summary and directions for future work.

[2] For further tempo induction evaluation details, on the database used here, we refer to [3].

2. DATA AND ASSOCIATED METADATA

The musical database we use for training and testing contains excerpts from 698 pieces of music, around 30 seconds long. The audio quality of this data is quite low, it was originally fetched in real audio format, with a compression factor of almost 22 with respect to the common 44.1 kHz 16 bits mono WAV format. It was subsequently converted to WAV format for experiments. This data is publically available on the world-wide web at the following URL:

http://www.ballroomdancers.com/Music/style.asp

Cha Cha	111 inst.	116 – 128 BPM
Jive	60 inst.	160 – 180 BPM
Quickstep	82 inst.	198 – 210 BPM
Rumba	98 inst.	90 – 110 BPM
Samba	86 inst.	96 – 104 BPM
Tango	86 inst.	120 – 140 BPM
Viennese Waltz	65 inst.	170 – 190 BPM
Slow Waltz	110 inst.	78 – 98 BPM

Table 1. Dance music classes, number of instances per class and class tempo ranges.

For all those recordings, the musical genre is available (see Table 1). In addition, the correct tempo, assigned manually, of each recording is also available (in beats per minute, BPM). The minimum value is 60 BPM, the maximum 224 BPM.

3. DESCRIPTORS

We consider 71 descriptors, divided into three groups. All are implemented as open source software under the GNU license.

3.1. Tempo

In addition to the ground truth tempo, in BPM, as provided with the data, we also computed the tempo using BeatRoot [1]. BeatRoot's tempo induction stage yields several tempo hypotheses that are subsequently refined, beat by beat, and ranked in a tracking process. The final tempo is the mean of the winning agent's inter-beat intervals.

The following 69 features describe low-level characteristics of 2 different periodicity representations (i.e. distribution statistics as centroid, flatness, etc. or peak-related quantities).

3.2. Periodicity Histogram descriptors

11 descriptors are based on a first representation of signal periodicities, the "periodicity histogram" (PH) [5]. This representation, is the collection in a histogram of the saliences of different pulses (from 40 BPM to 240 BPM) in successive chunks of signal (12s long, with overlap). In each chunk of signal, periodicities are computed via a comb filterbank.

3.3. Inter-Onset Interval Histogram descriptors

Remaining descriptors are quantities computed from a second representation of the signal periodicities, the Inter-Onset Interval Histogram (IOIH) proposed in [2]. This representation gives a measure of recurrence of the different inter-onset intervals present in the signal (not just successive onsets, but any pairs of onsets). Inter-onset intervals are accumulated in a histogram which is then smoothed by a Gaussian window.

We computed the saliences of 10 periodicities (prominent periods in the IOIH) whose periods are the 10 first integer multiples of the fastest pulse (computed as in [2]). Note that solely the period *salience* is kept, not the period value. Therefore, those descriptors are independent of the tempo.

We also defined 48 other descriptors as common "spectral" descriptors (8 distribution statistics and 40 MFCCs), but computed on the IOIH, not on a spectrum. The MFCC-like descriptors are computed as follows:

- IOIH computation
- Projection of the period axis from linear scale to the Mel scale, of lower dimensionality (i.e. 40), by means of a filterbank
- Magnitude logarithm computation
- Inverse Fourier transform

For each of the 30 classification tasks, we discarded the use of the tempo and we evaluated the relevances of the remaining low-level descriptors on an individual basis (i.e. Ranker search method associated to ReliefF attribute evaluator), and selected the 10 most relevant features. That is, the 30 classifiers all use 10 low-level features, that may be different in each case. All experiments have been conducted with Weka [6]. [3]

4. EXPERIMENTS

For classification, we use Support Vector Machines as it is commonly suggested for problems with few classes (especially 2). All percentages result from 10-fold cross-validation procedures. Systematic evaluation of different classification methods is left to future work.

The majority of the 28 pairwise classifier accuracies, all using 10 descriptors, are above 90%. The worst classifier is that between Slow Waltz and Viennese Waltz (81.8% accuracy, baseline 63%). The best is that between Quickstep and Viennese Waltz (100% accuracy, baseline 55.7%). Regarding the three-class classifiers, also using 10 descriptors, K_{29} (Quickstep vs. Rumba vs. Samba) has 84.1% accuracy (baseline 36.6%) and K_{30} (Cha Cha vs. Tango vs. Viennese Waltz) 91.9% accuracy (baseline 42.3%).

To measure the overall accuracy of the 30 classifiers, let us compute a weighted average of their individual accuracy. The weights are proportional to the number of times a classifier is actually required (given the tempo estimations of the 698 excerpts). This yields **89.4%** accuracy.

Let us now evaluate the whole classification process. Recall that the process involves two steps, it suffers from tempo estimation errors in addition to misclassifications. In 24.3% of the cases (i.e. 170 excerpts) the tempo estimation step assigns excerpts to pairwise (or three-class) classifiers that do *not* account for its true class. There is no way to recover from these errors, whatever the subsequent classification, the excerpt will be assigned to an incorrect class.

The overall accuracy of the system is therefore the multiplication of both step accuracies, i.e. $0.894 \times 0.757 =$ **67.6%**.

One might wonder whether considering metrical level errors in the design of the tempo probabilities (i.e. using tempo probabilitities as defined in Figure 2 instead of Figure 1) actually results in any improvement. As reported in [3], the tempo induction algorithm has around 50% accuracy (considering multiples as errors). The resulting overall accuracy of the method presented here would therefore be around $0.894 \times 0.5 = 44.7\%$. The improvement is over 20%.

However, we noted that tempo induction is especially bad for Slow Waltz excerpts, yielding around 75% to be assigned to wrong classifiers. This is because onset detection, in the tempo induction algorithm, is designed for percussive onsets, which are often lacking from waltzes. Removing the Slow Waltz excerpts for the database, 587

[3] *http://www.cs.waikato.ac.nz/ml/weka*

remain, and the number of excerpts that are assigned to irrelevant classifiers falls to 13.9%. The overall accuracy rises now to **76.5%**.

Those results are encouraging. They are however slightly lower than the results reported in [3], where the rationale was to build an 8-class classifier (1-Nearest Neighbour learner) with 15 MFCC-like descriptors and *no* tempo information. There, the classification accuracy reached **79%**.

5. SUMMARY, DISCUSSION AND FUTURE WORK

In this paper, we investigated the classification of 8 dance styles from a particular viewpoint that puts a special emphasis on the tempo estimation. The proposed classification process entails two subsequent steps: tempo computation and use of expert (pairwise or three-class) classifiers in specific tempo regions. We show that it is possible to design very accurate expert classifiers. However, logically, in this framework, if the tempo estimation fails, the classification fails. We showed that considering tempo multiples results in a substantial classification improvement.

The accuracy on a database of 698 excerpts from 8 classes is 67.6%. Restricting tests to the 7 classes (587 excerpts) on which tempo estimation is reasonably reliable, the accuracy is 76.5%. This is slightly worse than results reported in [3] with a different method (an 8-class classifier yielded 79% accuracy with 15 *tempo-independent* descriptors).

In conclusion, reducing the problem from an eight-class learning task to several two- or three-class learning tasks is only pertinent when using an *extremely* reliable tempo estimation algorithm.

To illustrate this, let us consider using the correct tempo (assigned manually) instead BeatRoot tempo (computed automatically). There, the classification accuracy rises to 82.1%. This corresponds to two factors: misclassifications of the expert classifiers (i.e. 0.109%) and the "cost" of the initial assumption regarding class tempo probabilities (i.e. instances —outliers— that effectively have a tempo outside of their class's tempo range, i.e. 54 of 698 instances).

This opens two important avenues for future work: improving the accuracy of the expert classifiers (for instance in refining the selection of the most relevant descriptors for each classifier) and study the validity of the limited-tempo-ranges assumption on a database containing more instances of a larger number of classes.

Further, as tempo-independent dance style classification [3] seems to be more reliable than tempo induction itself (that principally suffers from metrical level errors)— 79% vs. 50%—, we will study in future work whether genre classification can be used to improve tempo induction.

6. ACKNOWLEDGMENTS

For reading and very useful comments, we wish to thank Gerda Strobl, Elias Pampalk, Gerhard Widmer and (as always) Pedro Cano, Guenter Geiger and Perfecto Herrera.

The Austrian Research Institute for Artificial Intelligence also acknowledges the financial support of the Austrian Federal Ministries of Education, Science and Culture and of Transport, Innovation and Technology

This work was funded by the EU-FP6-IST-507142 project SIMAC (Semantic Interaction with Music Audio Contents). More information can be found at the project website *http://www.semanticaudio.org*.

7. REFERENCES

[1] Dixon S., "Automatic extraction of tempo and beat from expressive performances", Journal of New Music Research, 30 (1), 2001.

[2] Gouyon F., Herrera P. and Cano P., "Pulse-dependent analyses of percussive music", *Proc. AES 22nd International Conference*, Helsinki, 2002.

[3] Gouyon F., Dixon S., Pampalk E. and Widmer G. "Evaluating rhythmic descriptors for musical genre classification", *Proc. 25th AES International Conference*, London, 2004.

[4] Moelants D., "Dance music, movement and tempo preferences", *Proc. 5th Triennal ESCOM Conference*, Hanover, 2003.

[5] Pampalk E., Dixon S. and G. Widmer, "Exploring music collections by browsing different views", *Proc. International Conference on Music Information Retrieval*, Baltimore, 2003.

[6] Witten I. and Frank E., Data Mining: Practical machine learning tools with Java implementations, *Morgan Kaufmann*, San Francisco, 2000.