# Extending Deep Rhythm for Tempo and Genre Estimation Using Complex Convolutions, Multitask Learning and Multi-input Network

Hadrien Foroughmand Aarabi[1] and Geoffroy Peeters[2] *

[1] IRCAM - Sorbonne Université
had.foroughmand@gmail.com
[2] LTCI - Télécom Paris - Institut Polytechnique
geoffroy.peeters@telecom-paris.fr

**Abstract.** Tempo and genre are two inter-leaved aspects of music, genres are often associated to rhythm patterns which are played in specific tempo ranges. In this article, we focus on the Deep Rhythm system based on a harmonic representation of rhythm used as an input to a convolutional neural network. To consider the relationships between frequency bands, we process complex-valued inputs through complex-convolutions. We also study the joint estimation of tempo/genre using a multitask learning approach. Finally, we study the addition of a second input convolutional branch to the system applied to a mel-spectrogram input dedicated to the timbre. This multi-input approach allows to improve the performances for tempo and genre estimation.

**Keywords:** Tempo estimation, genre classification, deep-learning, complex network, multitask, multi-input.

## 1 Introduction

In the Music Information Retrieval (MIR) field, tempo is usually defined as the rate at which a listener taps while listening to a piece of music (Fraisse, 1982). The large number of works dedicated to its automatic estimation somehow demonstrates how important this task is for the MIR community, but also that there is still room for improving its estimation.

### 1.1 Related works

Rhythm as a musical concept represents all the temporal relations and information of an audio excerpt. It's definition is not formal, however several elements allows to define it such as the metrical structure, the timing and the tempo. The work on tempo estimation has for a long time focused on the development of

* This work was partly supported by European Union's Horizon 2020 research and innovation program under grant agreement No 761634 (FuturePulse project).

**hand-crafted systems**, often based on the perceptual process used in human tempo inference.

Several research in MIR are dedicated to the automatic analysis of the rhythmic elements. Among them, we can cite the ones of Longuet-Higgins and Lee (1982, 1984) on rhythmic perception and syncopation but also on the relationship between beat and meter induction (Honing and De Haas, 2008). One of the earliest system to estimate tempo proposed by Scheirer (1998) used a bank of band-pass filters followed by resonant comb-filters and a peak-picking process. Nearly a decade later, Klapuri et al. (2006) still used resonant comb-filters but as input to a process to track the rhythm at several metric levels. Gainza and Coyle (2011) developed a hybrid multi-band decomposition using auto-correlation of onset functions across multiple frequency bands. These works highlighted the strong relationship between tempo and beat tracking, since tempo can be estimated as the period between successive beats. Overviews of these systems can be found in (Gouyon et al., 2006; Zapata and Gómez, 2011; Peeters, 2011).

The appearance of large datasets annotated into tempo or beat/downbeat positions has favored the development of **data-driven systems** where the machine learns from the annotated data using machine-learning (ML) algorithms. The first ML algorithms used were K-Nearest-Neighbors (KNN) (Seyerlehner et al., 2007), Gaussian Mixture Model (GMM) (Xiao et al., 2008; Peeters and Flocon-Cholet, 2012), Support Vector Machine (SVM) (Chen et al., 2009; Gkiokas et al., 2012; Percival and Tzanetakis, 2014), bags of classifiers (Levy, 2011), Random Forest (Schreiber and Müller, 2017). Then deep learning (DL) became the most used ML algorithms in MIR. One of the first DL systems proposed for beat-tracking is the one of (Böck et al., 2015) which used resonant comb-filters applied to the output of a Reccurent Neural Network (Bi-LSTM) that predicts the beat position inside the raw audio and then estimates the periodicity as the predicted tempo. Later, Schreiber and Müller (2018) proposed the first end-to-end DL system (although starting from the mel-spectrogram) for tempo estimation. The mel-spectrogramm is used as input to a convolutional architecture that simulates a resonant comb filters. Their system considers the tempo prediction task as a classification task into tempo classes.

Starting from a rhythmic analysis in order to classify tracks into genres has also already been studied, notably by (Chew et al., 2005) in the case of ballroom dancing, but also by (Panteli et al., 2014) in the case of electronic dance music. Recently, Foroughmand and Peeters (2019) proposed to combine the two types of systems (hand-crafted and data-driven) in the so called "**Deep Rhythm**" (DR) system for tempo estimation and rhythm pattern/genre classification.

### 1.2 Deep Rhythm

Deep Rhythm is a system proposed in (Foroughmand and Peeters, 2019) which adapts a harmonic decomposition of rhythm to a deep learning formalism for tempo estimation and rhythm pattern classification. The method belongs to the data-driven systems in the sense that it uses machine learning to provide the ability to automatically learn and improve with programs that learn from these data. It also considers both the tempo and rhythm pattern in interaction by adequately modeling the audio content through a handcrafted feature representation. The tempo of a track can of course vary over time, but in this work a focus is made on the estimation of global tempo and the corresponding rhythm pattern. In (Foroughmand and Peeters, 2019), rhythm patterns referred to the genre classes of datasets that are composed of rhythmic (dances) classes such as Ballroom dataset.

*Harmonic decomposition of rhythm.* From Fourier series, it is known that any periodic signal $x(t)$ with period $T_0$ (or fundamental frequency $f_0 = 1/T_0$) can be represented as a weighted sum of sinusoidal components whose frequencies are the harmonics of $f_0$:

$$\hat{x}_{f_0,\underline{a}}(t) = \sum_{h=1}^{H} a_h \sin(2\pi h f_0 t + \phi_h) \tag{1}$$

For the voiced part of speech or pitched musical instrument, this leads to the so-called "harmonic sinusoidal model" (McAuley and Quatieri, 1986; Serra and Smith, 1990) that can be a starting point for audio coding or transformation. This model can also be used to estimate the pitch of a signal (Maher and Beauchamp, 1994): estimating the $f_0$ such that $\hat{x}_{f_0,\underline{a}}(t) \simeq x(t)$. The values $a_h$ can be estimated by sampling the magnitude of the DFT at the corresponding frequencies $a_{h,f_0} = |X(hf_0)|$. The vector $\underline{a}_{f_0} = \{a_{1,f_0} \cdots a_{H,f_0}\}$ represents the spectral envelope of the signal and is closely related to the timbre of the audio signal, hence the instrument playing. For this reason, these values are often used for instrument classification (Peeters, 2004).

For audio musical rhythm, Peeters (2006, 2010, 2011) proposes to apply such a harmonic analysis to an Onset Energy Function (OEF). The period $T_0$ is then defined as the duration of a beat (i.e. the time between two successive beats). In this harmonic analysis: $a_{1,f_0}$ then represents the DFT magnitude at the $4^{th}$-note level; $a_{2,f_0}$ at the $8^{th}$-note level; $a_{3,f_0}$ at the $8^{th}$-note-triplet level ... while: $a_{\frac{1}{2},f_0}$ represent the binary grouping of the beats $a_{\frac{1}{3},f_0}$ the ternary one

Peeters considers that the vector $\underline{a}$ is representative of the specific rhythm and that therefore $\underline{a}_{f_0}$ represents a specific rhythm played at a specific tempo $f_0$ (in this context, tempo is assimilated to the fundamental frequency). He proposes the following harmonic series: $h \in \{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{3}{4}, 1, 1.25, 1.33, \ldots, 8\}$.

An example of such harmonic decomposition of rhythm applied to a simplified signal is described in Fig. 1 [Left].

With the above-mentioned considerations, he shows:

- in (Peeters, 2011) that given the tempo $f_0$, the vector $\underline{a}_{f_0}$ can be used to classify different rhythm pattern;
- in (Peeters, 2006), that given manually-fixed prototype vectors $\underline{a}$, it is possible to estimate the tempo $f_0$ (looking for the f such that $\underline{a}_f \simeq \underline{a}$);
- in (Peeters, 2010) that the prototype vectors $\underline{a}$ can be learned (using simple machine learning) to achieve the best tempo estimation $f_0$.

The DR method is in the continuation of this last work: learning the values $\underline{a}$ to estimate the tempo or the corresponding rhythm pattern. It aims to adapt $\underline{a}$ to the deep learning formalism proposed by Bittner et al. (2017).

*Adaptation to a deep learning formalism.* In (Bittner et al., 2017), a task of fundamental frequency estimation in polyphonic music is achieved. To this aim, the depth of the input to a convolutional network is used to represent the harmonic series $\underline{a}_f$ and $f_0$ denotes the fundamental frequency. Bittner et al. (2017) propose in a first step to compute the Constant-Q Transform (CQT) of a harmonic signal.

The CQT is expanded to a third dimension which represents the harmonic series $\underline{a}_f$ of each $f$ (with $h \in [\frac{1}{2}, 1, 2, 3, 4, 5]$). When $f = f_0$, $\underline{a}_f$ will represent the specific harmonic series of the musical instrument (plus an extra value at the $\frac{1}{2}f$ position used to avoid octave errors). When $f \neq f_0$, $\underline{a}_f$ will represent (almost) random values.

The goal is to estimate the parameters of a filter such that when multiplied with this third dimension $\underline{a}_f$ it will provide very different values when $f = f_0$ or when $f \neq f_0$. This filter will then be convolved over all log-frequencies $f$ and time $\tau$ to estimate the $f_0$'s. This filter is trained using annotated data. In the method, there are actually several of such filters; they constitute the first layer of a CNN. In practice, in (Bittner et al., 2017), the $a_{h,f}$ are not obtained as $|X(hf)|$; but by stacking in depth several CQT each starting at different minimal frequencies i.e. by multiplying the lowest frequency in the range of the CQT $f_{min}$ by the $h$ coefficient: $hf_{min}$. A visual identification of $f_0$ is therefore possible when this transformation is applied on a simple harmonic signal by superimposing the CQT computed using the various values of $h$. The representation is denoted by Harmonic Constant-Q Transform (HCQT) of size $(f, \tau, h)$. An example of the HCQT applied to a harmonic signal is described in Fig. 1 [Right].

*Harmonic Constant-Q Modulation.* The goal of DR is to adapt the harmonic representation of the rhythm proposed in (Peeters, 2006, 2010, 2011) to the deep learning formalism proposed in (Bittner et al., 2017) in order to learn the weighting of the harmonic series relative to rhythm at each position $a_h$. The first step is to represent those harmonic series as a rhythmic representation named Harmonic Constant-Q Modulation (HCQM). For this, the HCQT proposed by Bittner et al. (2017) is not applied to the audio signal, but to a set of OEF which represent the rhythm content in several acoustic frequency bands. Each of those OEF is a low-pass signal whose temporal evolution is related to the tempo and the rhythm pattern, in this specific band.

As the Modulation Spectrum (MS) (Atlas and Shamma, 2003), which is a time/modulation-frequency representation, the HCQM represents the energy evolution (low-pass signal) within each acoustic frequency band $b$ of a first time/acoustic-frequency $(\tau/f)$. However, while the MS uses two interleaved STFT for this, a CQT is used for the second time/frequency representation in order to obtain a better spectral resolution. Finally, as proposed by Bittner

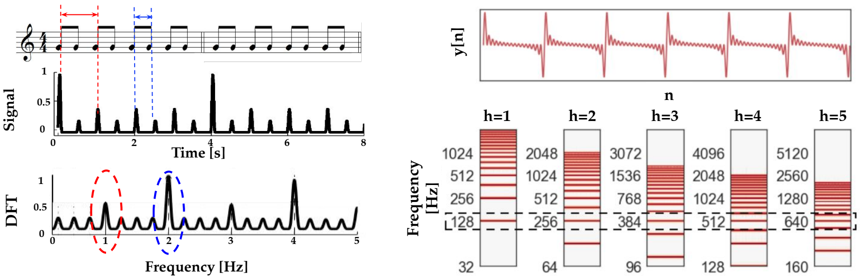

**Fig. 1.** **[Left]** Example of a harmonic representation of rhythm components of an onset energy signal. Each beat is divided into $8^{th}$-notes. The DFT of the OEF is represented at the bottom where vertical dashed lines represent $a_{h,f_0}$ with $h = 1, 2, 3, 4$ and the vertical dotted lines represent $a_{h,f_0}$ with $h = \frac{1}{4}, \frac{1}{2}, \frac{3}{4}$. Here, the signal have a tempo of 60BPM (1Hz) in red and a tactus of 120BPM (2Hz) in blue. Fig. inspired by Peeters (2011). **[Right]** Computation of the CQT of a harmonic signal according to the harmonic series $h$. The fundamental frequency is bordered by the black dotted rectangle. Fig. taken Bittner's PhD Thesis.

et al. (2017) and the HCQT representation, an extra dimension $h$ is considered to represent the content at the harmonics of each modulation frequency $\phi$.

The HCQM is finally a 4-dimensional representation of size $(\tau', \Phi, b, h)$ where $\tau'$ denotes the times of the CQT frames, $\phi$ the modulation frequencies (which correspond to the tempo frequencies), $b$ the acoustic frequency bands and $h$ the harmonic series. On Fig. 2, a visual identification of the possible tempo of a track on the HCQM representation is described.

*DR Convolutional Neural Network (CNN).* The architecture of the DR network is both inspired by the one from (Bittner et al., 2017) (since we perform convolutions over an input spectral representation and use its depth) and the one from (Schreiber and Müller, 2018) (since we perform a classification task). However, it differs in the definition of the input and output. In (Bittner et al., 2017), the input is the 3D-tensor $X_{hcqt}(f, \tau, h)$ and the convolution is done over $f$ and $\tau$ (with filters of depth $H$). In DR, the input could be the 4D-tensors of size $(\phi, \tau', b, h)$ and the convolution could be done over $\phi$, $\tau'$ and $b$ (with filters of depth $H$). However, to simplify the computation (in term of memory and computation time[3]), the input is reduced to to a sequence over $\tau'$ of 3D-tensors of size $(\phi, b, h)$. These inputs are denoted as $\tau'$-HCQM. The convolution is then done over $\phi$ and $b$ with filters of depth $H$.

The goal is to learn filters $W$ narrow in $\phi$ and large in $b$ which represent the specific shape of the harmonic content of a rhythm pattern. Convolution are pursued over $b$ because the same rhythm pattern can be played with instrument transposed in acoustic frequencies (lower or higher tuning). The first layer is followed by two convolutional layers of 64 filters of shape $(4, 6)$, one layer of 32

[3] The memory of the GPU servers is limited and 4D-convolutions are too costly.

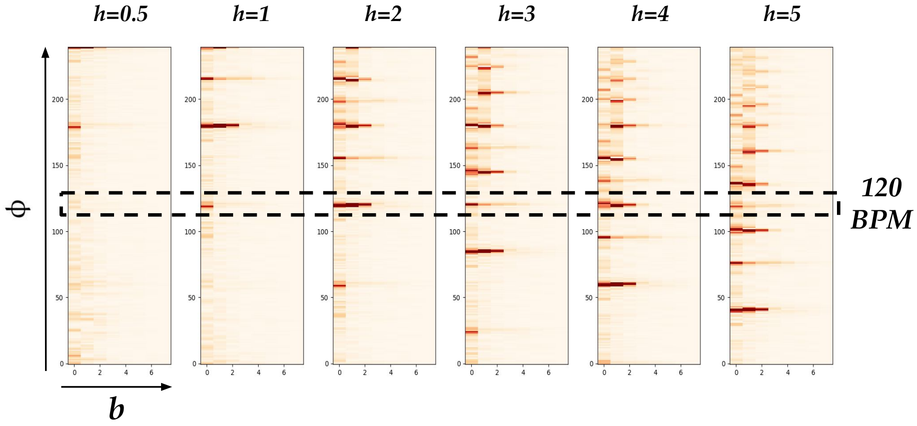

**Fig. 2.** HCQM example for a given temporal frame $\tau'$ of an audio excerpt. $\phi$ denotes the modulation frequency (associated with a candidate tempo), $b$ the acoustic frequency band and $h$ the harmonic coefficient. The tempo is visually identifiable at 120BPM by superimposing its rhythmic harmonic components through $h$.

filters of shape $(4, 6)$ and finally one layer of 8 filters of shape $(120, 6)$ (this allows to track down the relationships between the modulation frequencies $\phi$). As in (Schreiber and Müller, 2018), the tempo estimation problem is considered as a classification problem (instead of a regression one) into 256 tempo classes ranging from 30 to 286 BPM. This range is chosen to cover the different tempo values of the most popular music genres (and therefore those present in the majority of the state-of-the-art datasets) In DR, the outputs are either the $C = 256$ classes of tempo or the $C =$ genre classes number for genre classification. To do so, the output of the last convolution layer which is flattened and followed by a dropout with $p = 0.5$ (to avoid over-fitting (Srivastava et al., 2014)), a fully-connected layer of 256 units and the last fully-connected layer of $C$ units. All layers are preceded by a Batch Normalisation (BN) layer (Ioffe and Szegedy, 2015). Rectified Linear Units (ReLU) (Nair and Hinton, 2010) are used for all convolutional layers and Exponential Linear Units (ELU) (Clevert et al., 2016) for the first fully-connected layer. The architecture of the DR network is described in Fig. 3.

For the training, there are several $\tau'$-HCQM for a given track as input which are all associated with the same ground-truth (multiple instance learning). For the output dense layer, a softmax activation function is generally used for single label classification and is often associated with a categorical cross-entropy as loss-function between the predicted class $\hat{y}_c$ and the ground-truth $y_c$:

$$\mathcal{L} = -\sum_{c=1}^{C} y_c \log(\hat{y_c}) \tag{2}$$

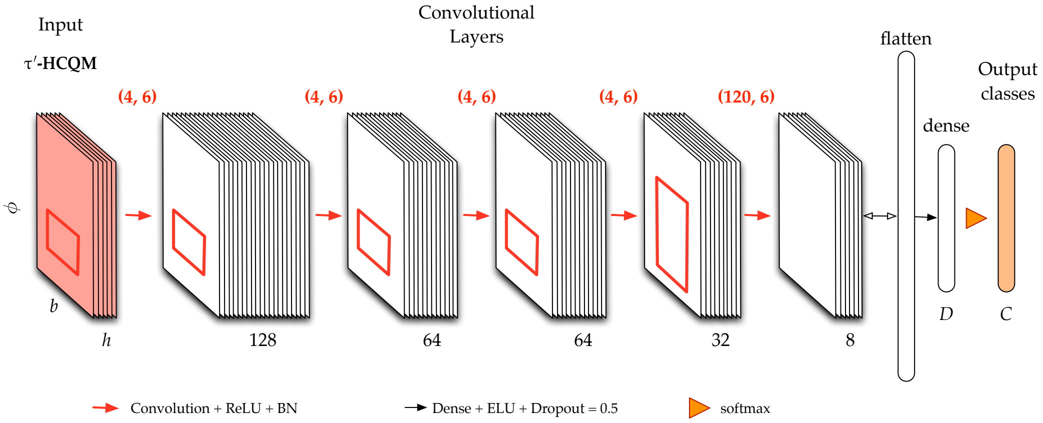

**Fig. 3.** DR model Architecture with $\tau'$-HCQM as input (of size $(\phi, b, h)$), the size of convolutional filters is indicated in red $(filter\_height, filter\_width)$, the number of kernel is indicated in black under each convolutional layer, $D$ denotes the number of units in the dense layer while $C$ is the number of units in the output layer (i.e. the classes logits).

where $c \in [1, C]$ refers either to the 256 tempo classes or to the genre classes. In this article, we assume that genres are estimated from methods based on rhythm analysis. We therefore deliberately choose not to make a distinction between rhythmic patterns and genres.

### 1.3 Proposals and article organisation

In this article, we present some extensions of the DR method. The development of each of these representations has as a starting point a musical intuition. For each of these extensions, we present our motivations and the resulting method.

*Complex Network.* In the original DR network, the input HCQM and the network do not allow to represent the inter-relationship between the various frequency bands $b$. This is due to the fact that each OEF is modeled by the modulus of the CQT and the modulus does not preserve the information of temporal location. Therefore, the network cannot consider the inter-relationship between the various acoustic frequency bands $b$. To face this, we propose to replace the use of the modulus of the HCQM by a complex-valued HCQM and turn it into an input of a complex CNN. This is described in section 2.

Recently, complex neural networks have appeared in the MIR field and allow model training on inputs with complex values. These models have proven their efficiency for various tasks such as automatic transcription (Trabelsi et al., 2017) of music or speech enhancement with a deep U-net (Choi et al., 2019). However, they have never been used for classification purposes.

*Multitask Learning.* In the original DR method, independent systems are trained for the task of tempo estimation and genre classification. We propose here a multitask approach where a single system is trained to solve both tasks simultaneously. This is done by defining two losses for the optimisation of the system. We believe that using a single network jointly trained for the two tasks would allow sharing information in the network. We want to exploit the rhythmic aspect common to both tasks so that each benefits from the other's learning. This is described in section 3.

Multitask learning methods have already shown their efficiency in various domains like computer vision (Girshick, 2015), natural language processing (Collobert and Weston, 2008) or speech recognition (Deng et al., 2013). In MIR field, they have been used for the estimation of the fundamental frequency (Bittner et al., 2018). In the case of rhythm description, the work of Böck et al. (2019) have showed good results by learning tempo estimation in parallel with the beat tracking.

*Multi-input Network.* The DR network was designed to represent the rhythm content of an audio track. As shown previously, the tempo range and possible rhythm patterns are strongly correlated to the music genre of the track. We believe that exploiting the benefits of other music-knowledge descriptors in addition to rhythm could improve the classification. We therefore study an extension of the DR by associating it with a second input branch. This branch is a network dedicated to the representation of timbre with a mel-spectrogram as input. It is based on a commonly-used network for audio tagging presented in (Choi et al., 2016). This is described in section 4.

*Evaluation.* Finally, we present the evaluation of the three methods in section 5 for the sake of comparison not only between them but also with the DR method. As these methods can be combined, the analysis of the results is also clearer.

## 2 Complex Deep Rhythm

A rhythmic pattern can be affiliated to simpler rhythm pattern (quarter note, eighth note, ...) played by different instruments at different frequency bands. Two different rhythm pattern examples are represented in Fig. 4.

① The bass drum and the snare drum are played simultaneously.

② The bass drum and the snare drum are played alternately.

In the original HCQM representation, the modulation frequencies $\phi$ are modeled independently by calculating the HCQT of the OEF in each acoustic frequency bands $b$. Since only the modulus of the HCQT is computed, the temporal information relative to the rhythmic components is not taken into account. The representation is then passed through a deep CNN. The network does not consider the inter-relationship between the various frequency bands $b$ when it learns

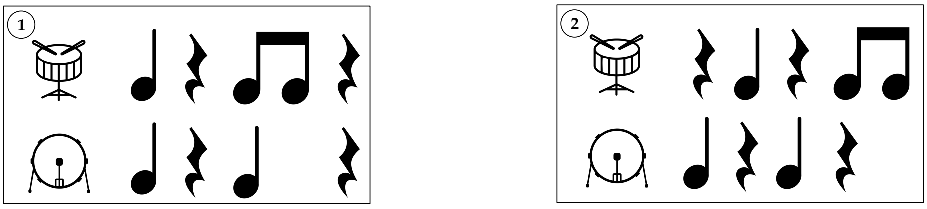

**Fig. 4.** Two examples of rhythm patterns.

the different rhythmic structure to perform the tempo (or the genre) estimation. For example, the snare drum and the bass drum being located on two different frequency bands, we can assume that the DR method is not able to differentiate between the two rhythmic patterns illustrated in Fig. 4. Moreover, ignoring these inter-band relationships impacts the learning of rhythmic components related to tempo within the HCQM.

In (Marchand and Peeters, 2014), the authors deal with the same limitation with their modulation scale spectrum representation. They therefore show in that modeling the inter-relationships between acoustic frequency bands using inter-band correlation coefficients allows to better estimate the rhythm pattern.

In our case, due to the data-driven aspect of the DR method, we would like to find a way to keep the temporal information of the rhythmic content present in each acoustic band $b$ of the HCQM and furthermore to be able to train a network that includes this information.

In a temporal representation of the frequency evolution such as the STFT or the CQT, the positional information of the windowed signals are contained in the phase of the complex-values.

We propose here to use a complex HCQM as input of a complex layer convolutional network (using the complex convolution described by Trabelsi et al. (2017)) in order to take into account the inter-relationships between the acoustic bands.

### 2.1 Complex HCQM

In order to obtain a complex representation of the HCQM, we modify one of its computation steps. When the HCQT is computed on the OEF of the STFT summed in acoustic frequency bands we keep the complex-values (in addition to the modulus). We therefore keep its real and imaginary parts instead of keeping only its absolute value. To be used as input of a CNN, the two parts are then superimposed on top of each other (Trabelsi et al., 2017), resulting in Cplx-$\tau'$-HCQM of a size ($\varPhi \times b \times 2h$).

### 2.2 Complex Convolution

The cplx-HCQM input to the layers is denoted by $H = H_{Re} + iH_{Im}$ (with $H_{Re}$ and $H_{Im}$ its real and imaginary parts, respectively). The complex kernel matrix of the layer (which is the trainable parameter) is denoted by $K = K_{Re} + iK_{Im}$ (with $K_{Re}$ and $K_{Im}$ its real and imaginary parts, respectively). The complex convolution is then expressed as:

$$K * H = (K_{Re} * H_{Re} - K_{Im} * H_{Im}) + i(K_{Im} * H_{Re} + K_{Re} * H_{Im}) \tag{3}$$

or expressed in matrix form as:

$$\begin{bmatrix} \Re(K * H) \\ \Im(K * H) \end{bmatrix} = \begin{bmatrix} K_{Re} & -K_{Im} \\ K_{Im} & K_{Re} \end{bmatrix} * \begin{bmatrix} H_{Re} \\ H_{Im} \end{bmatrix} \tag{4}$$

The output of each complex convolution layer is itself complex and is then used as input to the next complex convolution layer. All convolution layers of the original DR network are therefore replaced by complex convolution layers. Also, each complex convolution layers is followed by a complex-BN (as described in (Trabelsi et al., 2017)). After the last complex convolution, the resulting feature maps are flattened, hence by concatenating the real and imaginary outputs.

We illustrate this in Fig. 5 on which we only detail the complex convolution for the first convolution layer (the one applied to the input complex HCQM $H$). Instead of BN (Ioffe and Szegedy, 2015), which is applied only to real-values, we use complex-BN preceding each complex-convolution layer in order to ensure equal variance in both real and imaginary components. The whole complex-BN computation process is detailed in (Trabelsi et al., 2017).

On this Fig., the number of feature maps indicated under the layers are doubled compared to the original DR network since the number of complex-kernel are considered. For the implementation of the 2D complex convolution layers, we rely on the package *complexnn*[4] provided in (Trabelsi et al., 2017). We denote this method as Complex Deep Rhythm (Cplx-DR).

## 3 Multitask Learning

The goal of MultiTask Learning (MTL) is to share information between two related tasks in order to enable a model to generalise better on both of these tasks. The origin of MTL is biological. For example, according to our experience in learning to play tennis and squash, we find that the skill of playing tennis can help learn to play squash and vice versa. From a machine learning point of view, MTL can be seen as an inductive operation that can help learning a model by introducing an inductive bias. In this case, the inductive bias is provided by the auxiliary task which lead the model to favor hypotheses that are beneficial to all several tasks at once.

[4] `https://github.com/ChihebTrabelsi/deep_complex_networks`

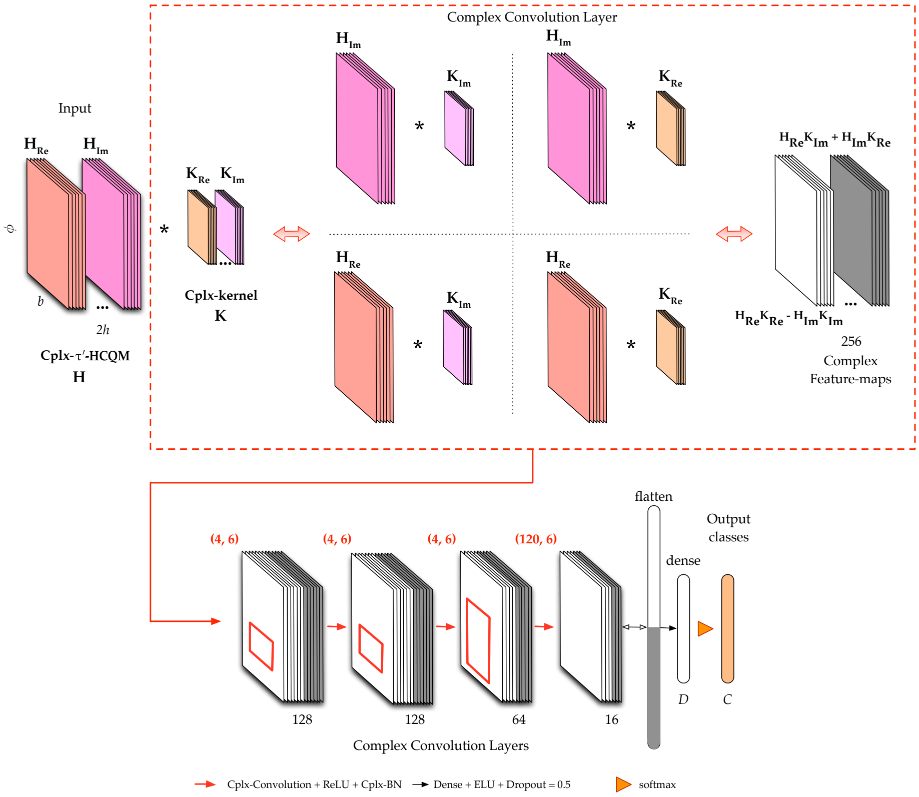

**Fig. 5.** Cplx-convolution applied to Cplx-$\tau'$-HCQM as input and Cplx-DR model Architecture. $D$ is the number of units in the dense layer, $C$ is the number of units in the output layer (i.e. the classes logits).

The classification of certain meta-genres such as Electronic Dance Music (EDM) and Ballroom Dances into sub-genres is mainly based on the analysis of their rhythmic structure since they are musical styles dedicated to dance. Thus it is possible to identify the sub-genre of one of these sounds by the tempo range at which it is played (in addition to their rhythmic patterns).

With the DR method (Foroughmand and Peeters, 2019), it is shown that the same network architecture can be used to achieve two different tasks: tempo estimation and genre classification. For each task, two different training and hence a different set of parameters are used. We want here to exploit this multitasking aspect through the implementation of a single network which jointly estimates the tempo and the rhythm pattern class.

Two main strategies stand out when it comes to MTL in the field of deep learning. These reside in the type of parameter sharing carried out between the hidden layers of the network.

When using the soft-sharing (Yang and Hospedales, 2016), each task has its own network. In order to learn similar parameters, the distance between them is regularised for each layer of the networks. With hard-sharing, the hidden layers are shared between all tasks while conserving specific layers for each task. This is the most common strategy and the one we chose to use. Hard parameter sharing has been proven to limit the risk of overfitting (Baxter, 1997).

Other benefits of using MTL in deep learning are to be considered. A model that jointly learn two tasks is able to learn a more general representation: it has an implicit data augmentation effect. It can be difficult for a model to learn the differences between relevant and irrelevant features, MTL acts as a focus on the various features since it combines the relevance of features from two linked tasks.

### 3.1 Multitask Deep Rhythm

*Architecture.* We illustrate the architecture of the network in Fig. 6. The convolutional part is the same as the original DR network (Foroughmand and Peeters, 2019). The extension starts at the flatten layer that follows the last convolutional layer. The associated vector then acts as input for two independent branches, each with two fully-connected layers ending with a softmax activation function. One branch is dedicated to genre classification, the other to tempo estimation. We choose $D = 64$ based on parameter validation experiments. The output of the first (resp. of the second) branch has the same size as the number of genre to be detected (resp. as the number of tempo classes).

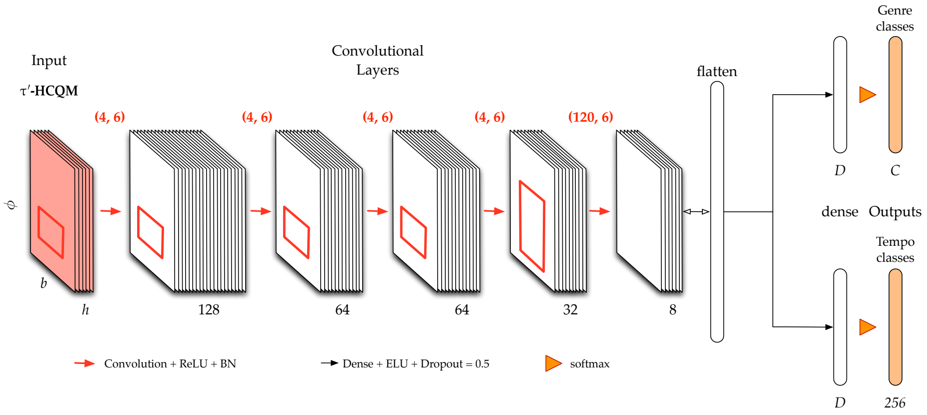

**Fig. 6.** MTL model Architecture with $\tau'$-HCQM as input (of size $(\phi \times b \times h)$), $D$ is the number of units in the dense layer, $C$ is the number of genre classes in an output layer and 256 tempo classes in the other.

It is important to note that using the MTL network architecture to deal with both tasks simultaneously allows to halve the number of trainable parameters.

Indeed, the original DR network is trained to perform the two tasks separately and therefore requires two independent training. The computation time remains quite similar between the DR and the MTL models training.

*Losses.* To train the system, we simultaneously minimise two categorical cross-entropy losses: one for the rhythm pattern classes $\mathcal{L}_{genre}$ and one for the tempo classes $\mathcal{L}_{tempo}$. Both are applied to the output of the sub-networks ended by a softmax activation function. Inspired by Bittner et al. (2018), we choose the use of an additive loss with equal weights between the genre loss and the tempo loss.

$$\begin{aligned}\mathcal{L} &= \mathcal{L}_{genre} + \mathcal{L}_{tempo} \\ &= -\sum_{c_g=1}^{C} y_{c_g} \log(\hat{y}_{c_g}) - \sum_{c_t=1}^{256} y_{c_t} \log(\hat{y}_{c_t})\end{aligned} \tag{5}$$

with the predicted genre class $\hat{y}_{c_g}$ and the genre ground-truth $y_{c_g}$ ($c_t$ for the tempo classes, respectively). We then minimise $\mathcal{L}$ (both losses $\mathcal{L}_{genre}$ and $\mathcal{L}_{tempo}$ are equally weighted) with the same ADAM optimiser as original DR method.

## 4 Multi-Input Network

The Deep Rhythm network was designed to represent the rhythm content of an audio track. As shown in (Gouyon et al., 2006) and demonstrated in our previous methods, the tempo range and possible rhythm patterns are strongly correlated to the music genre of the track. The DR network, however, focuses exclusively on the aspects related to rhythm, not on other features like instrumentation or timbre.

Since we want to perform a genre classification, our method could benefit from representations describing other musical elements. For instance in EDM, many genres were also defined, beyond their rhythmic structure, by timbre according to Butler (2006).

This intuition led us to the creation of a new network in order to introduce information distinct from rhythm when training the network for genre classification. To do so, we keep the convolutional part of the DR dedicated to rhythm with the HCQM as input and we add a convolutional branch dedicated to the representation of timbre and instrumentation with a log-mel magnitude spectrogram as input. We obtain a multi-input, multi-branch network and to denote this network we use the term multi-input network (and by extension Multi-Input MI method).

### 4.1 Method

The main inspiration of the additional branch is the work of Choi et al. (2016, 2017) who uses this type of convolutional layers with a mel-spectrogram as input to perform an automatic tagging task from large annotated databases. Pons et al. (2017) also recommend the use of log-mel magnitude spectrogram[5] as input of a CNN to analyse timbre patterns through the convolutional feature maps of the network.

*Mel-spectrogram.* Timbre is defined as the character or quality of a musical sound appart from its pitch, loudness and duration (Wessel, 1979). Also, music timbre is often associated with the identification of the instrument characteristics. It is related to the spectral envelope shape and the time variation of the spectral content (Peeters et al., 2011). Thus, it can be assumed that the mel-spectrogram is an adequate representation of timbre since it is a time/frequency expression well-suited to the human auditory system and so to the music perception.

Therefore, the mel-spectrogram contains timbre patterns that are considered as more pitch invariant than the STFT ones since they are based on a perceptual scale (Pons et al., 2017). For all these characteristics and also for the gain in performance it allows, this representation is one of the widespread features used in deep learning for various MIR tasks.

For tagging task, the mel-spectrogram is often used as input representation of a CNN trained with large-scale datasets. We choose to use a common architecture dedicated to the timbre branch of our MI network.

*Network Architecture* The first branch of the network is the one dedicated to rhythm with the HCQM as input followed by the DR convolutional layers. The second branch is the one dedicated to timbre characteristics followed by a network commonly-used for audio tagging with the mel-spectrogram as input. This latter branch is inspired from the well-known VGG network (Simonyan and Zisserman, 2015). Its architecture is adapted to large-scale analysis in the sense that its efficiency has been shown on large annotated databases. It is composed of a series of convolution layers associated with max-pooling layers.

The complete architecture of the MI network is described in Fig. 7. We take the same network parameters described in (Choi et al., 2016) and in (Choi et al., 2017). This network uses mel-spectrograms as input followed by five convolutional layers of $(3 \times 3)$ kernels[6] each connected to a max pooling layer $(2 \times 4), (2 \times 4), (2 \times 4), (3 \times 5), (4 \times 4)$ in order to reduce the size without losing information during training. In the original network, the last layer then predicts the tags. We skip it here. The output of the timbre branch does not need to be flatten since using max pooling already shapes the last layer as a $(1 \times 2048)$ vector. The flatten layer of the DR branch is concatenated with the last layer of the timbre branch and used as input of a dense layer of size 256. The output layer of the MI network is, as for the previous methods, the softmax activation

[5] For ease of reading we refer to it as mel-spectrogram

[6] In (Pons et al., 2017), it has been shown that the use of domain-knowledge inspired kernel size (e.g. by taking the whole frequency axis) leads to better performance in the case of large-scale training. However, because the difference in results is not significant on smaller scale analyses, so we choose to use square filters.

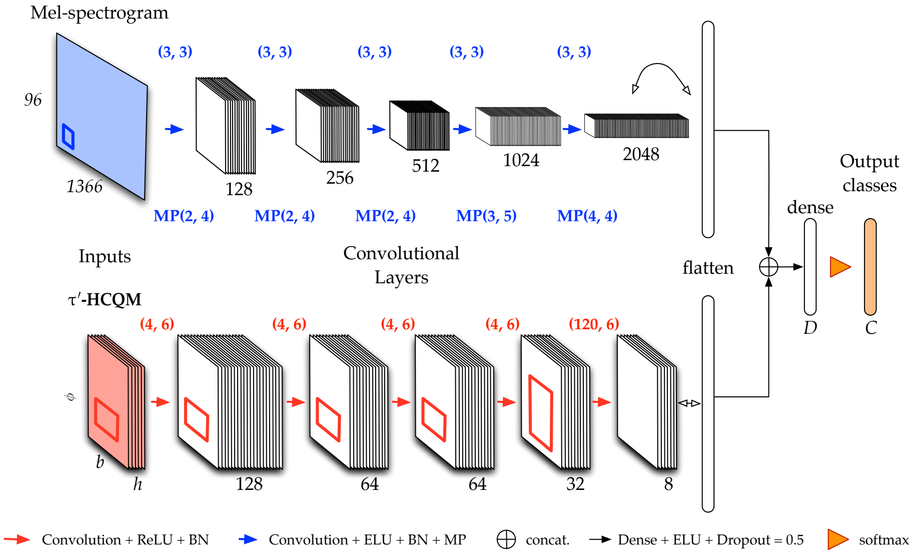

**Fig. 7.** MI model architecture. [Top] Branch dedicated to ”timbre”, VGG-like convolutional layers with a mel-spectrogram as input. [Bottom] Branch dedicated to rhythm, DR convolutional layers with an $\tau'$-HCQM as input.

function of $C$ classes ($C$ as the genre classes in the case of genre classification task or 256 tempo classes in the case of global tempo estimation). Again, the network is trained by minimizing a categorical cross-entropy.

## 5 Evaluation

### 5.1 Aggregating decisions over time

It is important to consider here that the DR network processes each temporal frame $\tau'$ independently. We denote by $x_{\tau'}$ the segment of the audio signal centered on time $\tau$ and of 8 s duration. During training, a Multiple Instance Learning paradigm is considered: each $x_{\tau'}$ is seen as an instance of the single global ground-truth tempo $T_{BPM}$ and the network trained accordingly. For testing, the output of the network (the softmax output) provides for each $x_{\tau'}$ a tempo likelihood vector $p(T_{BPM}|x_{\tau'})$ which represents the likelihood of each tempo $T_{BPM}$. The average over frame $\tau'$ of this vector is computed, $p(T_{BPM}) = \int_{\tau'} p(T_{BPM}|x_{\tau'})d\tau'$ and used to estimate the global tempo: $\hat{T}_{BPM} = \arg\max_{T_{BPM}} p(T_{BPM})$.

*Oracle Frame Predictor.* Can we find a better way to infer $T_{BPM}$ from the sequence of tempo likelihood vectors $p(T_{BPM}|x_{\tau'})$ ?. To answer this question, we would like to know what would be the upper bound achievable by DR to predict $T_{BPM}$ from the succession of $p(T_{BPM}|x_{\tau'})$. To do so we define an Oracle Frame

Predictor. This oracle knows which is the best frame $\tau'$ to be used to predict $T_{BPM}$. We denote this best frame by $\tau'^*$. The oracle defines the best frame as $\tau'^* = \arg\min_{\tau'}(T_{BPM} - \arg\max_{T_{BPM}} p(T_{BPM}|x_{\tau'})^2$. It is important to note that the final prediction of the oracle still uses the tempo likelihood vector to estimate the tempo (but only using the best frame): $\hat{T}^*_{BPM} = \arg\max_{T_{BPM}} p(T_{BPM}|x_{\tau'^*})$.

Typically, if the track only contains a single frame corresponding to $T_{BPM}$ and if the network is performing well, the Oracle should be able to find $\tau'^*$ and the corresponding $\hat{T}^*_{BPM}$ would be a good estimation. In the contrary, the average value $p(T_{BPM})$ will be blurred and $\hat{T}_{BPM} = \arg\max_{T_{BPM}} p(T_{BPM})$ would provide a wrong prediction. Hence $\hat{T}^*_{BPM}$ is an upper bound.

### 5.2 Tempo-only estimation

*Protocol.* To evaluate the performances on tempo estimation, we follow the same protocol as described in (Foroughmand and Peeters, 2019), i.e. we train the network on 3 datasets and evaluate the performances on 7 independent datasets. We also summarised the overall performances by indicating the results on the Combined dataset (the union of the 7 datasets). We indicate the results in terms of Accuracy1 ($($Acc1)) which is the accuracy considering a 4% tolerance window centered on the ground-truth global tempo value as correct estimation and Accuracy2 (*Acc2*) which is the estimation taking into account the predicted tempo at the second and the third octave above and below within a 4% window.

*Datasets.* For **Training** we used the following datasets:

- Extended Ballroom (EBR) (Marchand and Peeters, 2016) (3,826 tracks): an extension of the Ballroom (BR) dataset (Gouyon et al., 2006) with additional ballroom styles. For the tempo estimation task, we remove from the dataset the tracks already present in BR for experimental training/testing purposes.
- tempo MTG (tMTG): proposed by Faraldo et al. (2017) for EDM key estimation tempo annotated using a tapping method by Schreiber and Müller (2018) (1,159 tracks).
- tempo LMD (tLMD) : a subset of the Lack MIDI dataset proposed by Raffel (2016) and tempo annotated by Schreiber and Müller (2017) (3,611 tracks).

For **Testing** we used the following datasets:

- ACM: (Peeters and Flocon-Cholet, 2012) (1,410 tracks),
- ISMIR04: (Gouyon et al., 2006) (464 tracks),
- Ballroom (BR): (Gouyon et al., 2006) (698 tracks),
- Hainsworth (Hains.): (Hainsworth, 2004) (222 tracks),
- GTzan: (Marchand et al., 2015) (1,000 tracks),
- SMC: (Holzapfel et al., 2012) (217 tracks),
- Giantsteps (GST): (Knees et al., 2015) (664 tracks) and
- Combined: (4,675 tracks) the combination of all the test sets.

*Considered systems.* The results are indicated in Fig. 8. **DR**: original Deep Rhythm method; **Oracle-DR**: DR using the Oracle Frame Prediction; **Cplx-DR**: complex version of DR (section 2); **Oracle-Cplx-DR**: Oracle Frame Prediction of Cplx-DR. For comparison purposes, we parameterise the HCQM and the neural network in the same way as in (Foroughmand and Peeters, 2019) for the original DR method. Thus the modulation frequencies of the rhythmic representation correspond to the possible global tempo from 0 to 240 BPM and $h \in \{\frac{1}{4}, \frac{1}{3}, \frac{1}{2}, \frac{2}{3}, \frac{3}{4}, 1, 1.25, 1.33, \ldots, 8\}$.

*Results and discussion.* The first remark we can made is that the results of Oracle-Cplx-DR are better than those of Oracle-DR in terms of *Acc1* and *Acc2* for all independently evaluated datasets as well as for the combined one. This clearly demonstrates that the complex version allows a significant improvement in tempo estimation. It is necessary to mention here that we have verified in practice that these results were not simply due to the double size of the network convolutional layers. For this purpose, we have trained a non-complex network from the real-valued HCQM with the same kernel sizes as the complex network.

Another observation can be made at this stage by comparing the results obtained in terms of *Acc2*: we can see that DR and Cplx-DR, although slightly below, are almost at the same level as their Oracle version (except for SMC). This allows us to state that for a given track, the average of its predictions from its different $\tau'$-HCQM (respectively Cplx-$\tau'$-HCQM) reflects the presence of octave error in the estimated candidate when using the basic methods.

### 5.3 Joint tempo-genre, genre-only estimation

*Protocol.* It is not possible to perform the validation using cross-dataset validation. This is because genre classes are specific to each dataset. We only consider the datasets which are both annotated into tempo and into genre and perform for each a ten-fold cross validation (splitting each dataset into ten folds). For the tempo estimation, we indicate the same metric *Acc1* in % as above. We do not indicate the results in terms of *Acc2* here as we focus on joint learning and its comparison with the original DR method. For the genre classification, we indicate the mean-over-class recall $\hat{R}$ in % since it is independent of class distribution. The recall score for a class is the number of correctly detected items over the number of items in this class. The mean-over-class recall $\hat{R}$ is the average of all recall scores of each class.

*Datasets.* For the experiments, we used the following datasets each in a 10-fold cross-validation scenario:

- genre Extended Ballroom (gEBR): the selected genres are the same as in (Marchand and Peeters, 2016) in order to ensure a sufficient number of tracks per genre, the (3,992 tracks and 9 ballroom genres);
- Ballroom (BR): (Gouyon et al., 2006) (698 tracks and 8 genres);

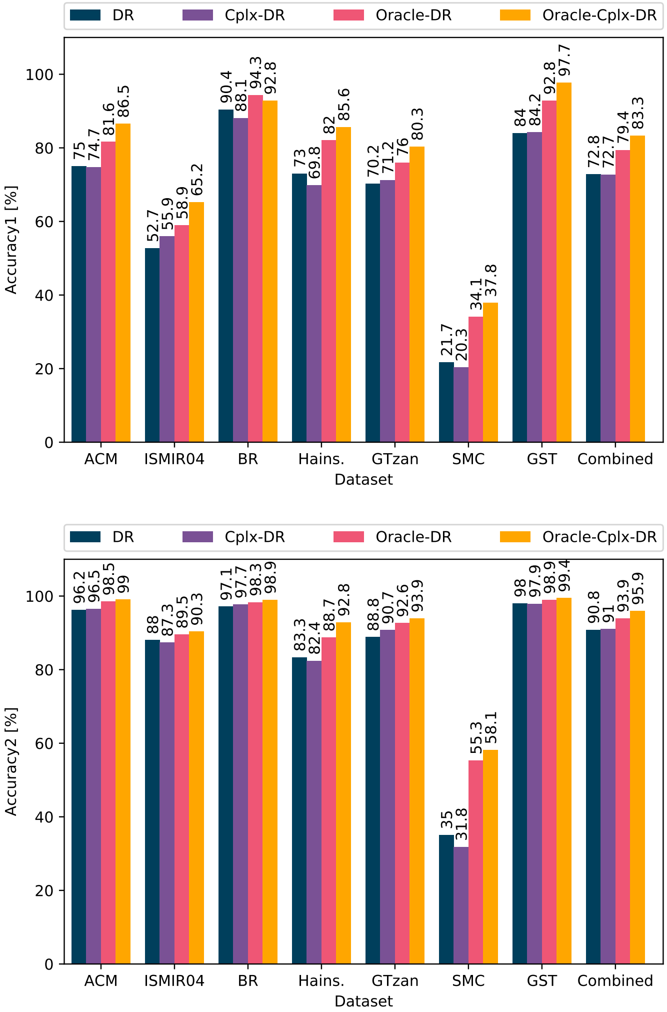

**Fig. 8.** Results of Cplx-DR and Oracle-Cplx-DR methods compared to the DR and Oracle-DR methods in terms of [Top] *Acc1* [Bottom] *Acc2*.

– genre MTG (gMTG): We propose to merge the two EDM datasets presented in subsection 5.2, GST and tMTG, keeping no duplicates. The goal is to obtain a tempo and genre annotated dataset for our experiments in both tasks (1,823 tracks and 23 electronic genres);

– GTzan (Marchand et al., 2015) (1,000 tracks and 10 various popular genres);

– Greek Dance (Gr) (Holzapfel and Stylianou, 2011) (180 tracks and 6 greek music genres not annotated in tempo).

*Considered systems.* Since the metrics are different we indicate the results in two tables: Tab. 1 for tempo estimation and Tab. 2 for genre classification.

- **DR**: Deep Rhythm network;
- **Cplx-DR**: complex version of DR;
- **MTL**: multitask learning (in section 3) (which simultaneously estimate the tempo indicated in Tab. 2 and the genre indicated in Tab. 1),
- **Cplx-MTL**: complex version of MTL;
- **MI**: multi-input network (section 4); (which can be used to estimate independently the tempo Tab. 2 or the genre Tab. 1),
- **Cplx-MI**: complex version of MI;
- **MI-MTL**: the multi-input multitask learning; (which estimate jointly the tempo Tab. 2 and the genre Tab. 1 independently),
- **Cplx-MI-MTL**: complex version of MI-MTL;
- For comparison purposes, we also provides the results with **CHOI** model (Choi et al., 2017) using the same protocol.

**Table 1.** Comparative and joint estimation results of genre classification in term of mean-over-class recall $\hat{R}$ for all methods.

| Dataset | CHOI | DR | Cplx-DR | MTL | Cplx-MTL | MI | Cplx-MI | MI-MTL | Cplx-MI-MTL |
|---|---|---|---|---|---|---|---|---|---|
| BR | 60.1 | 93.0 | 86.5 | 92.1 | 86.1 | **94.2** | 92.3 | 93.0 | 91.9 |
| gEBR | 72.1 | 95.2 | 92.1 | 94.8 | 92.4 | **96.5** | 93.9 | 96.2 | 94.6 |
| Gr | 38.1 | 68.9 | 40.0 | - | - | **69.4** | 47.2 | - | - |
| gMTG | 21.7 | 37.6 | 36.4 | 37.1 | 39.8 | 37.3 | **40.6** | 39.6 | 40.3 |
| GTzan | 74.2 | 59.1 | 43.5 | 57.1 | 44.0 | **74.3** | 74.1 | 67.2 | 66.0 |

**Table 2.** Comparative and joint estimation results of global tempo estimation in term of Accuracy1 for all methods.

| Dataset | CHOI | DR | Cplx-DR | MTL | Cplx-MTL | MI | Cplx-MI | MI-MTL | Cplx-MI-MTL |
|---|---|---|---|---|---|---|---|---|---|
| BR | - | 92.8 | 90.0 | **93.2** | 91.4 | 91.3 | 92.7 | 92.2 | 92.4 |
| gEBR | - | 95.4 | 95.7 | 96.4 | 95.6 | **96.1** | 94.6 | 96.0 | 95.7 |
| gMTG | - | 91.3 | 91.8 | 91.2 | **92.0** | 91.6 | 90.1 | 91.3 | 91.6 |
| GTzan | - | 72.4 | 72.4 | **74.8** | 70.8 | 73.3 | 69.5 | 71.5 | 68.5 |

*Results and discussion.* Regarding the genre classification, we noticed that the results of Cplx-DR are lower to those obtained with the DR method.

The results of the MTL and Cplx-MTL methods presented in the two tables for tempo estimation and genre classification show that joint learning of the both tasks is justified. Indeed, the network uses half as many parameters than DR or MI to perform the two tasks simultaneously and to obtain these results despite the fact that they perform slightly below the other methods for gEBR and gMTG. For BR the best results are achieved with the MTL method. Results for genre classification are similar compared to the other methods for the various test sets (except for GTzan). We conclude that the MTL method allows both estimations to benefit from each other, showing that the genres are strongly characterised by the global tempo range of the evaluated tracks.

We can also observe that the results obtained with the DR method are much better than the CHOI baseline for all datasets except for GTzan. We can thus conclude that a small-scale rhythm-oriented classification method is much more efficient than a method more generally applied to large-scale dataset tagging. The results on GTzan can be explained by the fact that the CHOI method has shown good results in previous work on dataset balanced in various popular music genres. As learning takes place on this various data, it allows a VGG-like model to generalise better.

We can observe that the MI method is the most efficient for genre classification of BR, gEBR, Gr, GTzan. Moreover, for the MTG dataset it is the Cplx-MI version that performs best. Although learning is joint, the MI-MTL methods fail to match the previous ones in terms of statistical results. Thus, we can conclude that a model combining different input music-informed features of a CNN is more efficient for genre estimation.

## 6 Conclusion

In this article, we presented three main extensions of the Deep Rhythm method (which adapts a harmonic decomposition of rhythm to a deep learning formalism for tempo estimation and rhythm pattern classification) with the aim of exploiting the specific characteristics of this method to perform tempo estimation and genre classification. This method shows that using music-informed features as input of a data-driven system is efficient for such MIR tasks.

First, we wanted to take into account inter-band acoustic relationships in order to improve our estimation. To do this, we proposed to integrate the temporal relations between the bands through the combined learning of the module and the phase of our HCQM representation by a neural network. We kept the complex-values when calculating the Cplx-HCQM used as input of a complex network using complex convolution layers. The effectiveness of this complex version of the DR, the Cplx-DR, is shown through its evaluation for the tempo estimation task. This method allows a clear improvement of the results when analyzing the results obtained via the Oracle Frame Prediction in terms of *Acc1*.

Second, in order to better take into account the interdependences between tempo and genre we proposed a multitask network where the two tasks of tempo and genre estimation are jointly solved. For this, we proposed a hard parameter sharing network architecture with shared hidden layers and with two independent output layers dedicated to each task. This network was trained to minimise the additive categorical cross-entropy losses of the two outputs. We showed that MTL led to an improvement for both tasks on the only evaluated EDM dataset gMTG. In addition, even if the results are not better than the ones of the previous methods, it is important to emphasise that they are obtained using a single network trained to perform both tasks.

Third, we wanted to take into account an other descriptor in addition to the rhythm one represented by the HCQM and the DR network. We put forward a multi-branch/multi-input network where VGG-like convolutional layers with mel-spectrogram input are added to represent timbre information. We showed that this MI architecture allowed a much better genre classification for almost all evaluated datasets.

As future work, it might be interesting to use the HCQM as input to a MTL method that estimate other rhythmic element such as beats or downbeats in addition to tempo. Furthermore, regarding the results of the MI method, it could be interesting to investigate the effects of other music-related features (in addition to rhythm) like pitch or other features of timbre on genre classification.

# Bibliography

Atlas, L. and Shamma, S. A. (2003). Joint acoustic and modulation frequency. *Advances in Signal Processing, EURASIP Journal on*, 2003:668–675.

Baxter, J. (1997). A bayesian/information theoretic model of learning to learn via multiple task sampling. *Machine learning*, 28(1):7–39.

Bittner, R. M., McFee, B., and Bello, J. P. (2018). Multitask learning for fundamental frequency estimation in music. *arXiv preprint arXiv:1809.00381.*

Bittner, R. M., McFee, B., Salamon, J., Li, P., and Bello, J. P. (2017). Deep salience representations for f0 estimation in polyphonic music. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Suzhou, China.

Böck, S., Davies, M. E., and Knees, P. (2019). Multi-task learning of tempo and beat: Learning one to improve the other. In *Proc. of ISMIR (International Society for Music Information Retrieval).*

Böck, S., Krebs, F., and Widmer, G. (2015). Accurate tempo estimation based on recurrent neural networks and resonating comb filters. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Malaga, Spain.

Butler, M. J. (2006). *Unlocking the groove: Rhythm, meter, and musical design in electronic dance music.* Indiana University Press.

Chen, C.-W., Cremer, M., Lee, K., DiMaria, P., and Wu, H.-H. (2009). Improving perceived tempo estimation by statistical modeling of higher-level musical descriptors. In *Audio Engineering Society Convention 126.* Audio Engineering Society.

Chew, E., Volk, A., and Lee, C.-Y. (2005). Dance music classification using inner metric analysis. In *The next wave in computing, optimization, and decision technologies*, pages 355–370. Springer.

Choi, H.-S., Kim, J.-H., Huh, J., Kim, A., Ha, J.-W., and Lee, K. (2019). Phase-aware speech enhancement with deep complex u-net. *arXiv preprint arXiv:1903.03107.*

Choi, K., Fazekas, G., and Sandler, M. (2016). Automatic tagging using deep convolutional neural networks. *Proc. of ISMIR (International Society for Music Information Retrieval).*

Choi, K., Fazekas, G., Sandler, M., and Cho, K. (2017). Convolutional recurrent neural networks for music classification. In *Proc. of IEEE ICASSP (International Conference on Acoustics, Speech, and Signal Processing)*, pages 2392–2396, New Orleans, USA. IEEE.

Clevert, D.-A., Unterthiner, T., and Hochreiter, S. (2016). Fast and accurate deep network learning by exponential linear units (elus). *International Conference on Learning Representations.*

Collobert, R. and Weston, J. (2008). A unified architecture for natural language processing: Deep neural networks with multitask learning. In *Proceedings of the 25th international conference on Machine learning*, pages 160–167.

Deng, L., Hinton, G., and Kingsbury, B. (2013). New types of deep neural network learning for speech recognition and related applications: An overview.

In *Proc. of IEEE ICASSP (International Conference on Acoustics, Speech, and Signal Processing)*, pages 8599–8603, Vancouver, British Columbia, Canada. IEEE.

Faraldo, A., Jorda, S., and Herrera, P. (2017). A multi-profile method for key estimation in edm. In *Audio Engineering Society Conference: 2017 AES International Conference on Semantic Audio*. Audio Engineering Society.

Foroughmand, H. and Peeters, G. (2019). Deep-rhythm for tempo estimation and rhythm pattern recognition. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Delft, The Netherlands.

Fraisse, P. (1982). Rhythm and tempo. *The psychology of music*, 1:149–180.

Gainza, M. and Coyle, E. (2011). Tempo detection using a hybrid multiband approach. *Audio, Speech and Language Processing, IEEE Transactions on*, 19(1):57–68.

Girshick, R. (2015). Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448.

Gkiokas, A., Katsouros, V., and Carayannis, G. (2012). Reducing tempo octave errors by periodicity vector coding and svm learning. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Porto, Portugal.

Gouyon, F., Klapuri, A., Dixon, S., Alonso, M., Tzanetakis, G., Uhle, C., and Cano, P. (2006). An experimental comparison of audio tempo induction algorithms. *Audio, Speech and Language Processing, IEEE Transactions on*, 14(5):1832–1844.

Hainsworth, S. W. (2004). *Techniques for the Automated Analysis of Musical Audio*. PhD thesis, University of Cambridge, UK.

Holzapfel, A., Davies, M. E., Zapata, J. R., Oliveira, J. L., and Gouyon, F. (2012). Selective sampling for beat tracking evaluation. *Audio, Speech and Language Processing, IEEE Transactions on*, 20(9):2539–2548.

Holzapfel, A. and Stylianou, Y. (2011). Scale transform in rhythmic similarity of music. *Audio, Speech and Language Processing, IEEE Transactions on*, 19(1):176–185.

Honing, H. and De Haas, W. B. (2008). Swing once more: Relating timing and tempo in expert jazz drumming. *Music Perception*, 25(5):471–476.

Ioffe, S. and Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proc. of ICML (International Conference on Machine Learning)*, pages 448–456.

Klapuri, A. P., Eronen, A. J., and Astola, J. T. (2006). Analysis of the meter of acoustic musical signals. *Audio, Speech and Language Processing, IEEE Transactions on*, 14(1):342–355.

Knees, P., Faraldo, A., Herrera, P., Vogl, R., Böck, S., Hörschläger, F., and Le Goff, M. (2015). Two data sets for tempo estimation and key detection in electronic dance music annotated from user corrections. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Malaga, Spain.

Levy, M. (2011). Improving perceptual tempo estimation with crowd-sourced annotations. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Miami, Florida, USA.

Longuet-Higgins, H. C. and Lee, C. S. (1982). The perception of musical rhythms. *Perception*, 11(2):115–128.

Longuet-Higgins, H. C. and Lee, C. S. (1984). The rhythmic interpretation of monophonic music. *Music Perception*, 1(4):424–441.

Maher, R. C. and Beauchamp, J. W. (1994). Fundamental frequency estimation of musical signals using a two-way mismatch procedure. *JASA (Journal of the Acoustical Society of America)*, 95(4):2254–2263.

Marchand, U., Fresnel, Q., and Peeters, G. (2015). GTZAN-Rhythm: Extending the GTZAN Test-Set with Beat, Downbeat and Swing Annotations. Late-Breaking Demo Session of the 16th International Society for Music Information Retrieval Conference, 2015.

Marchand, U. and Peeters, G. (2014). The modulation scale spectrum and its application to rhythm-content description. In *Proc. of DAFx (International Conference on Digital Audio Effects)*, pages 167–172, Erlangen, Germany.

Marchand, U. and Peeters, G. (2016). The extended ballroom dataset. In *Late-Breaking/Demo Session of ISMIR (International Society for Music Information Retrieval)*, New York, USA. Late-Breaking Demo Session of the 17th International Society for Music Information Retrieval Conf.. 2016.

McAuley, R. and Quatieri, T. (1986). Speech analysis/synthesis based on a sinusoidal representation. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 34:744–754.

Nair, V. and Hinton, G. E. (2010). Rectified linear units improve restricted boltzmann machines. In *Proc. of ICML (International Conference on Machine Learning)*, pages 807–814, Haifa, Israel.

Panteli, M., Bogaards, N., Honingh, A. K., et al. (2014). Modeling rhythm similarity for electronic dance music. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, pages 537–542, Taipei, Taiwan.

Peeters, G. (2004). A large set of audio features for sound description (similarity and classification) in the cuidado project. Cuidado project report, Ircam.

Peeters, G. (2006). Template-based estimation of time-varying tempo. *Advances in Signal Processing, EURASIP Journal on*, 2007(1):067215.

Peeters, G. (2010). Template-based estimation of tempo: using unsupervised or supervised learning to create better spectral templates. In *Proc. of DAFx (International Conference on Digital Audio Effects)*, pages 209–212, Graz, Austria.

Peeters, G. (2011). Spectral and temporal periodicity representations of rhythm for the automatic classification of music audio signal. *Audio, Speech and Language Processing, IEEE Transactions on*, 19(5):1242–1252.

Peeters, G. and Flocon-Cholet, J. (2012). Perceptual tempo estimation using gmm-regression. In *Proceedings of the second international ACM workshop on Music information retrieval with user-centered and multimodal strategies*, pages 45–50. ACM.

Peeters, G., Giordano, B. L., Susini, P., Misdariis, N., and McAdams, S. (2011). The timbre toolbox: Extracting audio descriptors from musical signals. *JASA (Journal of the Acoustical Society of America)*, 130(5):2902–2916.

Percival, G. and Tzanetakis, G. (2014). Streamlined tempo estimation based on autocorrelation and cross-correlation with pulses. *Audio, Speech, and Language Processing, IEEE/ACM Transactions on*, 22(12):1765–1776.

Pons, J., Slizovskaia, O., Gong, R., Gómez, E., and Serra, X. (2017). Timbre analysis of music audio signals with convolutional neural networks. In *2017 25th European Signal Processing Conference (EUSIPCO)*, pages 2744–2748. IEEE.

Raffel, C. (2016). *Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching*. PhD thesis, Columbia University.

Scheirer, E. D. (1998). Tempo and beat analysis of acoustic musical signals. *JASA (Journal of the Acoustical Society of America)*, 103(1):588–601.

Schreiber, H. and Müller, M. (2017). A post-processing procedure for improving music tempo estimates using supervised learning. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Suzhou, China.

Schreiber, H. and Müller, M. (2018). A single-step approach to musical tempo estimation using a convolutional neural network. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Paris, France.

Serra, X. and Smith, J. (1990). Spectral modeling synthesis: A sound analysis/synthesis system based on a deterministic plus stochastic decomposition. *Computer Music Journal*, 14(4):12–24.

Seyerlehner, K., Widmer, G., and Schnitzer, D. (2007). From rhythm patterns to perceived tempo. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Vienna, Austria.

Simonyan, K. and Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. *International Conference on Learning Representation*.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Trabelsi, C., Bilaniuk, O., Zhang, Y., Serdyuk, D., Subramanian, S., Santos, J. F., Mehri, S., Rostamzadeh, N., Bengio, Y., and Pal, C. J. (2017). Deep complex networks. *arXiv preprint arXiv:1705.09792*.

Wessel, D. L. (1979). Timbre space as a musical control structure. *Computer music journal*, pages 45–52.

Xiao, L., Tian, A., Li, W., and Zhou, J. (2008). Using statistic model to capture the association between timbre and perceived tempo. In *Proc. of ISMIR (International Society for Music Information Retrieval)*, Philadelphia, PA, USA.

Yang, Y. and Hospedales, T. M. (2016). Trace norm regularised deep multi-task learning. *arXiv preprint arXiv:1606.04038*.

Zapata, J. and Gómez, E. (2011). Comparative evaluation and combination of audio tempo estimation approaches. In *Audio Engineering Society Conference: 42nd International Conference: Semantic Audio*. Audio Engineering Society.